nature communications

Article

https://doi.org/10.1038/s41467-024-53303-4

Deep learning resilience inference for complex networked systems

Received: 18 November 2023

Accepted: 8 October 2024

Published online: 24 October 2024

Check for updates

Chang Liu [1,2,5], Fengli Xu [1,2,5], Chen Gao [1,2], Zhaocheng Wang [1,2], Yong Li [1,2] ✉ & Jianxi Gao [3,4] ✉

Resilience, the ability to maintain fundamental functionality amidst failures and errors, is crucial for complex networked systems. Most analytical approaches rely on predefined equations for node activity dynamics and simplifying assumptions on network topology, limiting their applicability to real-world systems. Here, we propose ResInf, a deep learning framework integrating transformers and graph neural networks to infer resilience directly from observational data. ResInf learns representations of node activity dynamics and network topology without simplifying assumptions, enabling accurate resilience inference and low-dimensional visualization. Experimental results show that ResInf significantly outperforms analytical methods, with an F1-score improvement of up to 41.59% over Gao-Barzel-Barabási framework and 14.32% over spectral dimension reduction. It also generalizes to unseen topologies and dynamics and maintains robust performance despite observational disturbances. Our findings suggest that ResInf addresses an important gap in resilience inference for real-world systems, offering a fresh perspective on incorporating data-driven approaches to complex network modeling.

Complex systems in diverse domains, including ecology, biochemistry, and physiology, are commonly characterized as networked systems with interconnected nodes and weighted links[1–4]. Resilience[5–7], defined as the capacity to maintain functionality under perturbations, is a fundamental property of these systems. The foundational insights into resilience can be traced back to Robert May's seminal work[8], where he pioneered the investigation of stability equilibrium in these networked systems. In 1973, Holling[9] conceptualized the notion of resilience as the degree of external perturbations a system can endure, but he provided limited insights on the empirical measurements. Later studies[10] further analyzed the responses of networked systems to perturbations, evaluating the connectivity in the context of node and edge disruptions. The concept of network resilience was formally defined recently[5], articulating that a resilient system should invariably converge to a desired, non-trivial stable equilibrium after perturbation. Prior research[11,12] indicates that the loss of resilience often has extensive implications. For instance, in ecological systems, the decline in resilience can precipitate mass species extinctions[13]. This underscores the profound consequences of resilience loss and highlights the urgent need for approaches to predict and mitigate such outcomes.

Generally, given a complex networked system $G = (\mathbf{V}, \mathbf{A})$ with N nodes, we use $\mathbf{V}$ to represent the node set and $\mathbf{A}$ to represent the weighted interaction matrix, where $A_{ij} \in \mathbb{R}$ denotes the interaction intensity between node i and j. The activity of a specific node i is represented as x_i, governed by the following non-linear differential equation:

$$\frac{dx_i}{dt} = \mathrm{F}(x_i) + \sum_{j=1}^{N} A_{ij}\mathrm{G}(x_i, x_j), \tag{1}$$

where $\mathrm{F}(x_i)$ on the right side signifies the self-dynamics of node i, $\mathrm{G}(x_i, x_j)$ characterizes the interaction dynamics between node i and j.

[1]Department of Electronic Engineering, Tsinghua University, Beijing, P. R. China. [2]Beijing National Research Center for Information Science and Technology (BNRist), Tsinghua University, Beijing, P. R. China. [3]Department of Computer Science, Rensselaer Polytechnic Institute, Troy, NY, USA. [4]Network Science and Technology Center, Rensselaer Polytechnic Institute, Troy, NY, USA. [5]These authors contributed equally: Chang Liu, Fengli Xu.
✉e-mail: liyong07@tsinghua.edu.cn; jianxi.gao@gmail.com

The solution pattern of stable equilibrium for Equation (1) describes system resilience. The system is resilient if there is a non-zero unique stable equilibrium that indicates its functional state. Otherwise, it is non-resilient if there are stable dysfunctional equilibria that can trap the system or the system lacks the stable equilibrium, exhibiting chaotic behavior.

Noteworthy contributions[5,14–17] have sought to provide analytical estimates for resilience of *N*-dimensional systems with complex interactions between components by reducing them to tractable one-dimensional systems based on mean-field theory, spectral graph theory, etc. Despite their groundbreaking successes, these approaches often make strong assumptions about network topology and dynamics for analytical feasibility[18]. We use the classic Gao-Barzel-Barabási (GBB)[5] framework as an example to explore the constraints of these assumptions. Specifically, GBB computes a single resilience parameter $\beta_{\mathrm{eff}} \in \mathbb{R}$ for a networked system. The system is deemed resilient only if its resilience parameter exceeds a certain critical threshold, i.e., $\beta_{\mathrm{eff}} > \beta^{c}_{\mathrm{eff}}$ (Fig. 1d–f). However, the accurate estimation of β_{eff} and β^{c}_{eff} relies on the strong assumptions that node activity dynamics can be described with linear equations and the degrees of interconnected nodes are almost mutually independent. We evaluate GBB on several synthetic networked systems governed by representative mutualistic[19], gene regulatory[3], and neuronal dynamics[20] (Dynamics and datasets in Methods). In all three settings, we observe that inaccurate inferences consistently occur in networked systems with positive or negative assortativity. Taking the systems with mutualistic dynamics as examples, they are both inferred to be non-resilient due to their $\beta_{\mathrm{eff}} < \beta^{c}_{\mathrm{eff}}$ (Fig. 1d). However, the networked system with negative assortativity is, in fact, resilient (Fig. 1a, red color), because its network activity consistently converges to a single equilibrium (Fig. 1g, red curve). These findings demonstrate that classic analytic models often yield inaccurate inferences when their foundational assumptions are violated. Therefore, accurately inferring network resilience remains an unresolved challenge in practice, largely due to the divergence between the simplifying assumptions of current analytical frameworks and the complexities encountered in real-world contexts.

In this study, we demonstrate how deep learning methods can effectively leverage the increasingly available observational data to extend the idea of resilience inference to real-world complex networked systems. We design a powerful deep learning framework that effectively integrates Transformer and Graph Neural Network (GNN) architectures, facilitating the learning of Resilience Inference

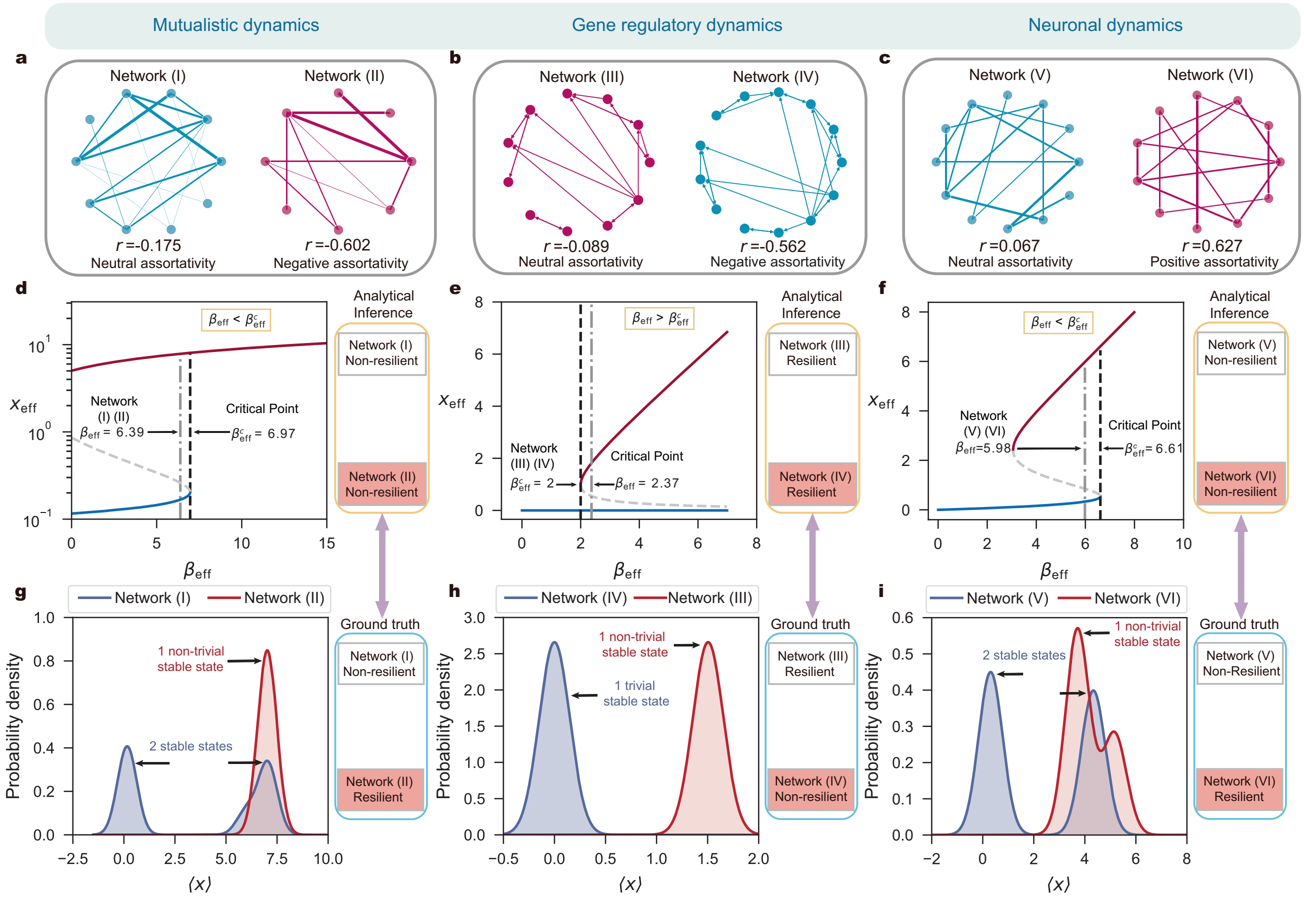

Fig. 1 | Example cases of simplified analytical inferences. We perform case studies on networks with mutualistic (parameter setting: $B = 0.1, C = 1, K = 5, D = 5, E = 0.9, H = 0.1$) (**a**), gene regulatory (parameter setting: $f = 1, h = 2, B = 1$) (**b**), and neuronal (parameter setting: $\mu = 3.5, \delta = 2$) (**c**) dynamics. Specifically, we first plot the 1-D resilience function condensed by GBB from the original high dimensional equations, where we show the stable equilibrium of x_{eff} (Supplementary Equation (11)) given β_{eff} (Supplementary Equation (12)). We analyze pairs of networks that have similar β_{eff} under GBB framework but different assortativity measured by degree correlation coefficient *r*. Network (I),(II),(V), and (VI) are all inferred as non-resilient by the classic GBB framework because $\beta_{\mathrm{eff}} < \beta^{c}_{\mathrm{eff}}$ (**d**, **f**). On the contrary, Network (III) and (IV) are inferred as resilient with $\beta_{\mathrm{eff}} > \beta^{c}_{\mathrm{eff}}$ (**e**). We further simulate the node activities using the corresponding dynamics for 100 times with varying initial conditions (Dynamics and datasets in Methods) on each network, and employ kernel density estimation (KDE) plot to visualize the distribution of its stable states. However, the ground truth simulations show Network (II), (III) and (VI) are resilient because they have a unique, non-trivial stable states ($\langle x \rangle > 0$), while Network (I), (IV) and (V) are non-resilient for having more than 1 stable states or only 1 trivial stable state ($\langle x \rangle = 0$) (**g**–**i**). Therefore, Network (II), (IV), (VI) are inaccurately inferred by the classic analytic model GBB, and they all have negative or positive assortativity.

https://doi.org/10.1038/s41467-024-53303-4

representations (ResInf for short) across various complex networked systems. Specifically, ResInf employs stacked Transformer encoder layers[21] to generate representations for the governing equations of node activity dynamics by modeling the complex correlations among node activities, which is subsequently merged with topological representations through GNN's message passing mechanisms[22–26]. The learned representations for multiple observations are aggregated and projected into a specialized 1-dimensional decision space, termed '*k*-space'. This space enables precise classification of resilient systems and provides insights on the systems' closeness to critical thresholds that indicate shifts in resilience status.

We employ ResInf on real-world microbial systems in a laboratory setting where traditional analytical approaches are infeasible due to the unavailability of definitive governing equations for node activity dynamics. Our numerical results demonstrate that ResInf achieves an impressive accuracy rate, attaining an F1-score of up to 0.829 on average. Additionally, when assessed on synthetic networked systems driven by three representative equations, i.e., mutualistic, gene regulatory, and neuronal dynamics, ResInf markedly outperforms its conventional counterparts, achieving a maximum F1-score improvement of up to 41.59% and 14.32% compared to Gao-Barzel-Barabási (GBB) framework[5] and spectral dimension reduction (SDR) approach[14], respectively. If we train ResInf on data covering different patterns of resilience loss, including phase shifts and the emergence of alternative stable states, it can effectively capture these underlying patterns and leverage such generalizable knowledge to infer the resilience of systems with previously unseen dynamics. Importantly, ResInf maintains robustness under various types of observation noise, including missing or spurious links in network topologies and inaccuracies in node activity trajectories. These findings demonstrate that carefully designed deep learning models can effectively exploit observational data for complex system analysis and suggest that the proposed ResInf framework can be adapted to diverse real-world complex systems without the crutch of oversimplifying assumptions.

Results

ResInf framework

Here we formulate the resilience inference problem investigated in our work. We denote a complex networked system with N nodes as $G = (\mathbf{V}, \mathbf{A})$ with node set $\mathbf{V}$ and adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$, where $A_{ij} \in \mathbb{R}$ denotes the interaction intensity between node i and j. Its node activities at time step t are denoted as $\mathbf{x}(t) \in \mathbb{R}^N$. The resilience of the networked system, i.e., the label, can be represented by a binary variable $y \in \{0, 1\}$. The system is resilient ($y = 1$) if node activities consistently converge to a non-zero and unique equilibrium regardless of initial node activity conditions. Otherwise, the system is non-resilient ($y = 0$) if there are stable dysfunctional equilibria that can trap the system or it lacks the stable equilibrium, only exhibiting chaotic behavior. Based on this definition, we can determine labels of system resilience through the observation of node activities without knowing underlying dynamics equations, i.e., Equation (1) of the system. Our objective is to infer the resilience y of the system in advance from a data-driven perspective. The input data comprises system topology, i.e., $\mathbf{A} \in \mathbb{R}^{N \times N}$, which describes interactions between components, and node activities $\mathbf{X} \in \mathbb{R}^{M \times N \times d}$ containing $\mathbf{x}(t)$ of M observed trajectories with the first d initial steps where the resilience of system remains unknown to us. Each trajectory is initiated with a unique node activity condition.

To address the problem, we present a deep learning Resilience Inference framework (ResInf, Fig. 2). Specifically, ResInf leverages stacked transformer networks[21] as a dynamics encoder to learn dense representations for the underlying node activity dynamics from the input node activity trajectories $\mathbf{X}$ without any prior knowledge (Supplementary Fig. 1)[21,27]. Different trajectories are encoded in parallel, thus the module produces M node activity dynamics representations for each node. Moreover, we design a topology encoder that uses graph neural network (GNN)[22] to model the non-Euclidean network topology from the input adjacency matrix $\mathbf{A}$ with a message-passing mechanism that recursively aggregates features from neighboring nodes (Supplementary Fig. 2). It can generate discriminating topological representations for each node's multi-hop neighborhood[28]. We use node activity dynamics representations from the dynamics encoder module as the input of the topology encoder module, which treats them as the initial node features and incorporates them with topology information via the message passing GNN layers, thereby incorporating information of node activity dynamics to the produced topological representations. Subsequently, these learned representations are transmitted to and consolidated by a virtual global node, providing a holistic system representation describing system-level characteristics. Owing to the parallel processing of different trajectories in aforementioned procedures, we obtain M system representations, each capturing knowledge from an input node activity trajectory and integrating information from network topology. Given that network resilience is characterized by achieving a consistent stable equilibrium across different trajectories, we develop a trajectory aggregator designed to harness M system representations derived from different trajectories. It uses the self-attention network to assign weights of M system representations dynamically and fuses them to a final system representation accordingly (Supplementary Fig. 3). The fused system representation is then projected to a 1-dimensional k-space with the fully connected network, which facilitates informative visualization and accurate resilience inference. We illustrate more technical details of ResInf in Methods.

Resilience inference of real-world microbial systems

Microbial systems are ubiquitous in the real-world and essential for organic decomposition and nutrient cycling. Therefore, their resilience significantly contributes to the ecological balance. We employ the empirical data collected by a recent study that investigates dynamic species compositions in bacterial microcosms[29], where abundances of species and competition interplay between them correspond to node activities and network topology, respectively. Each data contains abundances of species from the same microbial community observed via laboratory experiments. Resilience of these systems are determined by fluctuation intensity of observed species abundances, which is consistent with the existing work[29] (Real-world microbial systems in Methods). Species abundances of non-resilient microbial systems with intense competition often exhibit chaotic fluctuations, as opposed to that of resilient systems marked by a stable equilibrium[30]. We illustrate examples of the relative species abundances of resilient and non-resilient microbial systems in Fig. 3a. Real-world microbial systems often exhibit complex intrinsic activity dynamics among diverse types of bacteria[29,31], rendering them inapplicable for analytical approaches since such behavior cannot be concisely described by equations but can only be assessed through observations. Therefore, we employ our data-driven approach ResInf on microbial systems as a case study to validate its real-world performance. For each system, we input species abundances to the model along with a complete network with uniform edge weight as the default topology, following the maximum entropy principle. We randomly split the real-world microbial systems into training and test sets (85% vs 15%).

We compare ResInf with several representative data-driven baselines, including two deep learning architectures—the multilayer perceptron model (MLP), long short-term memory model (LSTM)[32], and two graph kernel-based models—shortest path kernel model (SP)[33] and propagation kernel model (PROP)[34] (Supplementary Note 1.1 ~ 1.4). We utilize the F1-score (Evaluation metrics in Methods), a combined metric of precision (how many inferred resilient systems are actually resilient) and recall (how many resilient systems are correctly inferred)

https://doi.org/10.1038/s41467-024-53303-4

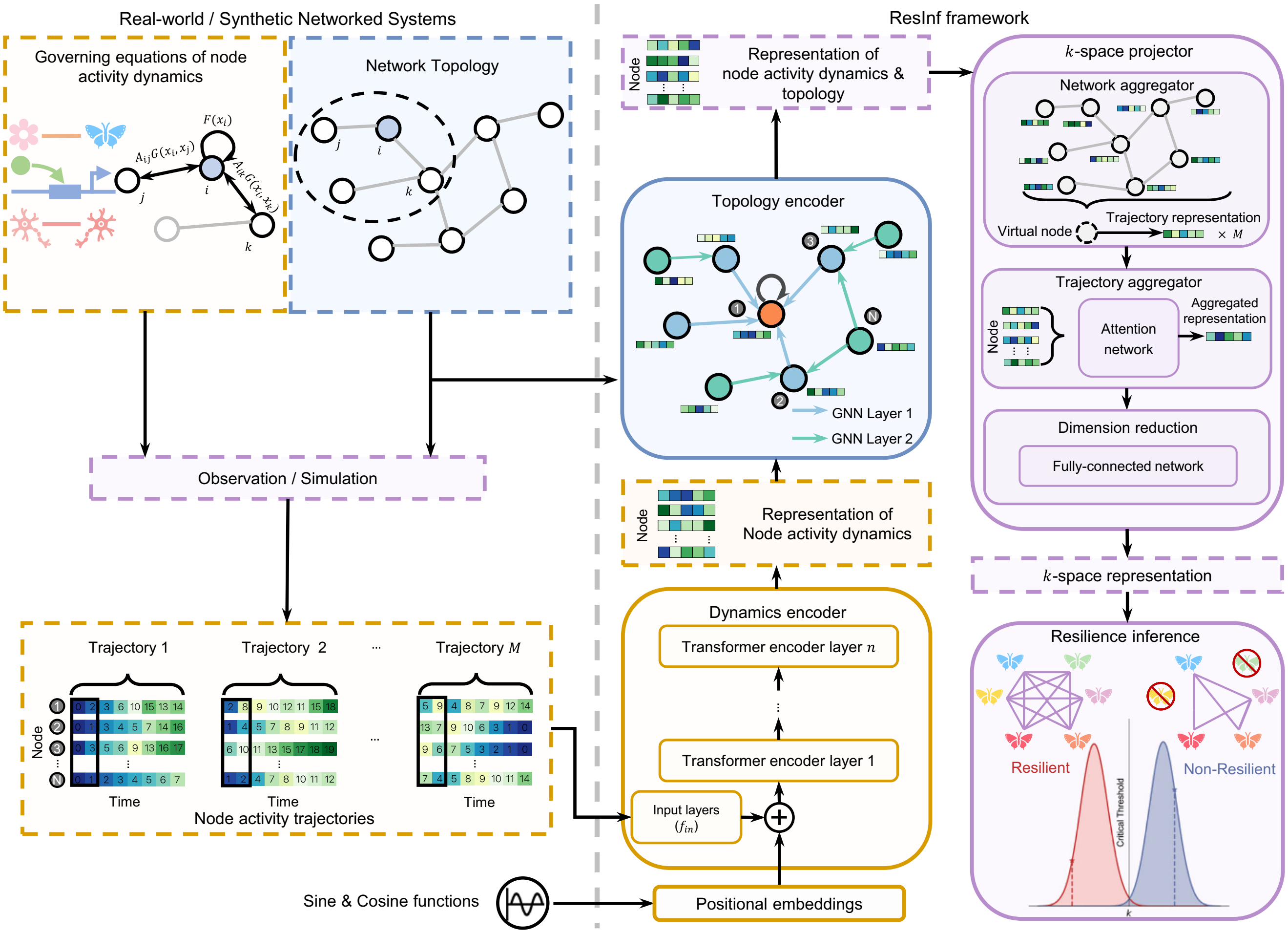

Fig. 2 | The proposed ResInf framework. ResInf leverages real-world observed or simulated network topologies and node activity trajectories to infer the resilience of networked systems without any prior assumption. It contains three modules: the dynamics encoder maps node activity trajectories to dense representations with stacked Transformer encoder layers, which models the complex correlations between node activities and produces representations for node activity dynamics; topology encoder leverages graph neural network to generate discriminating topological representations for each node's multi-hop neighborhoods; *k*-space projector aggregates the representations for node activity dynamics and topologies via a virtual global node, and employs a multi-head self-attention network to fuse the representations learned from various trajectories dynamically. Subsequently, it uses a dimension reduction network to project the aggregated representation to a 1-dimensional *k*-space, facilitating accurate resilience inference with linear classifiers.

to quantify the performances of models. Figure 3b demonstrates that ResInf can achieve a high F1-score of 0.829 ± 0.028 in inferring the resilience of real microbial systems, which surpasses competitive baselines LSTM by 34.14%. It shows that deep learning architectures without specific designs for resilience inference are usually sub-optimal for the problem, which can also apply to graph kernel-based models SP and PROP that regard node activities as static features neglecting their temporal dependencies. Conventional analytical frameworks (for example, GBB) are even not applicable in this scenario since we cannot describe the system with explicit dynamics equations to find a theoretical bifurcation point. The results demonstrate that our data-driven deep learning framework, ResInf, effectively extends resilience inference to real-world microbial systems without prior knowledge and assumptions, holding promising potential for resilience inference across various networked systems even if their node activity dynamics are complex and difficult to describe explicitly.

In broader scenarios where real-world systems have not yet collapsed and only limited training data are available from the system under investigation, directly training ResInf becomes impractical. To avoid collecting large amounts of training data in the wild or laboratories, we propose an alternative approach that leverages simulated data to train ResInf, where node activity trajectories are generated by solving dynamics equations on synthetic topologies. We use these microbial systems as a representative test dataset to validate our approach. Specifically, we evaluate the performances of models when trained solely on simulated data from SIS dynamics (Dynamics and datasets in Methods) and subsequently apply them to infer the resilience of these real-world microbial systems (Fig. 3c, details in Supplementary Note 2). The results demonstrate that ResInf achieves a competitive F1-score of 0.807 ± 0.016. Notably, this performance shows slight differences from the F1-score of 0.829 ± 0.028 obtained when the framework is trained directly on laboratory-collected data from real microbial systems. It underscores the practical applicability of ResInf in real-world scenarios, eliminating the need and effort for extensive data collection from both laboratories and real environments.

Resilience Inference of Synthetic Networked Systems

To evaluate the performance of ResInf across various scenarios, we further conduct resilience inference experiments on synthetic networked systems governed by three representative dynamics, i.e., mutualistic, gene regulatory, and neuronal dynamics. Specifically, we use the network topologies empirically collected from ecosystems and cells for mutualistic and regulatory systems[5,35], respectively. Besides,

https://doi.org/10.1038/s41467-024-53303-4

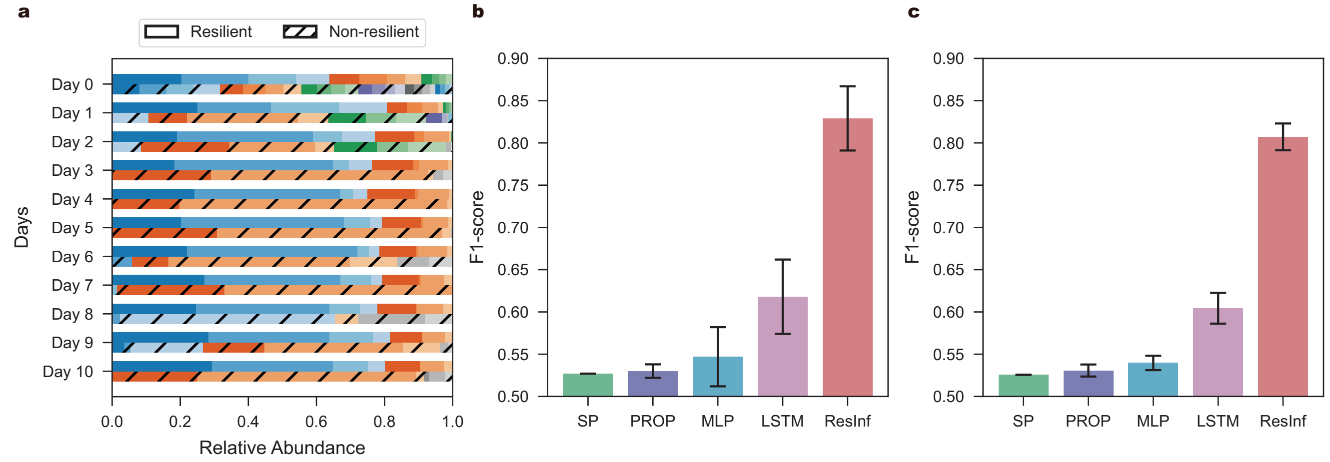

Fig. 3 | Resilience inference of real-world microbial systems. Data of each microbial system comprises abundances of species in the same microbial community observed during 11 days of laboratory experiments[29]. The resilience of systems (i.e., labels) are determined by the fluctuation intensity of species abundances (the coefficient of variation, Real-world microbial systems in Methods). **a** Relative species abundances of resilient (without black slash) and non-resilient (with black slash) microbial systems. Colors are employed to distinguish between different species. Species abundances of the non-resilient microbial system display persistent fluctuation and cannot converge to a stable equilibrium. **b** F1-scores of different models trained on real-world microbial systems. We employ species abundances of the first three days along with a complete network with uniform edge weight as the default topology (following the maximum entropy principle) for both training the model and inferring the resilience of microbial systems. We randomly split microbial systems into training and test sets (85% vs 15%). **c** F1-scores of different models trained on simulated systems (Supplementary Note 2). It should be noted that conventional analytical frameworks (for example, GBB) are not applicable in this scenario since we cannot describe the system with explicit dynamics equations. Bars represent the average values, and error bars represent the standard deviation ($n = 35$ with different random seeds).

we use the classic Erdős-Rényi[36] and Barabási-Albert[37] models to generate topologies for systems governed by neuronal dynamics, which is consistent with the setting in previous study[38]. Moreover, it is of great importance to examine the resilience inference performance in the scenarios in which the networked systems experience specific perturbations. Therefore, we follow the settings in existing works[5] to perform random perturbations of node removal, link removal, and global weight changes on the synthetic networked systems (Simulating critical events in Methods). Finally, the proposed ResInf and baseline models are trained on a randomly sampled subset of the synthetic networked systems, and evaluated on a test set with previously unseen networked systems. Besides the aforementioned four data-driven frameworks, including MLP, LSTM, SP, and PROP, we also introduce two conventional models—Gao-Barzel-Barabási (GBB)[5] and spectral dimension reduction (SDR)[14] frameworks (Supplementary Note 1.5 ~ 1.6). GBB offers a dimension reduction approach that effectively reduces an N-dimensional system into a one-dimensional degree-related β-space, subject to the constraint of neutral assortativity topology. SDR utilizes the dominant eigenvalue α of the system topology to achieve accurate dimension reduction grounded in a theoretical premise. These two approaches are exemplary methods for estimating resilience of networked systems from the analytical perspective.

The experiment results demonstrate ResInf consistently outperforms all baseline models in F1-score across networked systems governed by mutualistic, gene regulatory, and neuronal dynamics (Fig. 4a–c). For example, ResInf registers an F1-score of 0.977 ± 0.005 on networked systems with mutualistic dynamics, while the F1-scores of classic analytical model GBB and SDR are 0.690 ± 0.015 and 0.858 ± 0.007, respectively, despite GBB and SDR leverage explicit node activity dynamics equations. Therefore, based on just a few initial observed node activities about investigated systems compared to existing analytical frameworks, the superior performance of ResInf indicates that it effectively captures the underlying resilience loss patterns and the complex interplay between topology and node activity dynamics from observed data, shedding lights on the potential of learning-based approaches in understanding complex networked systems. ResInf also outperforms all data-driven baselines, i.e., LSTM, PROP, SP, and MLP. It is worth pointing out that LSTM and PROP have comparable performance with GBB and SDR, but only ResInf significantly outperforms GBB and SDR in all settings ($p < 0.01$, two-sided Student's t-test). Besides, we find that ResInf consistently has the best performance on each subset of the networked systems classified by the aforementioned perturbation types (Supplementary Fig. 5) and in different hyper-parameter settings (Supplementary Note 4, Supplementary Figs. 12–13). Furthermore, we evaluate our framework ResInf on synthetic data with low and high assortative topologies (Supplementary Fig. 6 and Supplementary Tables 6–7). The results reveal that the performance of GBB is inferior on high-assortative networks compared to low-assortative networks. Although SDR shows some improvement, there still remains a noticeable gap in its performance on networks with mutualistic dynamics. On the contrary, our ResInf framework demonstrates consistent performance across both types of networks, with no discernible difference in its efficacy. Beyond the application to mutualistic, gene regulatory, and neuronal networks, our developed framework can also explore the resilience of other systems with different node activity dynamics and structural properties (see Supplementary Note 3 and Supplementary Fig. 7 for human interactions with epidemic spreading cases). The above results demonstrate that ResInf can make accurate resilience inference across networked systems with various settings, representing an important methodological advance that outperforms the state-of-the-art analytical model without leveraging explicit dynamics equations.

Generalizability of ResInf

The networked systems under consideration may emerge in new contexts where node activity dynamics parameters, or even the underlying equations themselves, differ from existing data, thus hindering the application of analytical approaches that rely on comprehensive knowledge of systems' node activity dynamics. Furthermore, the lack of labels for these new systems also impedes our ability to directly retrain the models and extend their applicability to such novel contexts. In this section, we assess the out-of-sample inference performance of ResInf, demonstrating its generalizability across both

https://doi.org/10.1038/s41467-024-53303-4

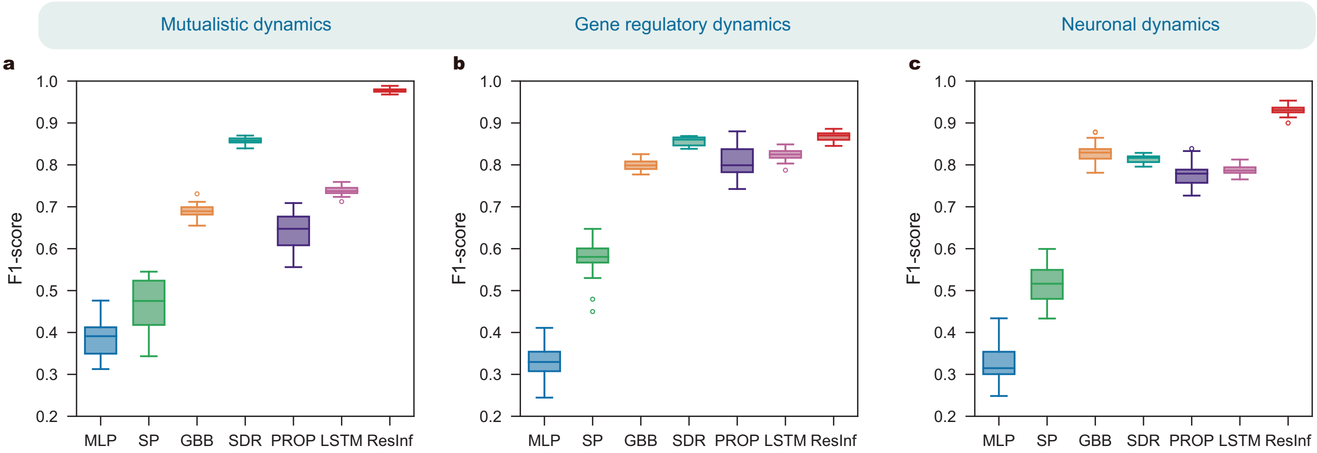

Fig. 4 | Resilience inference performance on synthetic networked systems. We evaluate resilience inference performance of ResInf along with other baseline methods on synthetic network datasets. These datasets include mixed types of perturbed networks of mutualistic (**a**), gene regulatory (**b**), and neuronal (**c**) dynamics, respectively (Supplementary Table 2). The parameter of each dynamics is the setting outlined in index 1 of Supplementary Table 4. The dataset of each dynamics contains 2000 samples, which are partitioned into training, validation, and test sets at a ratio of 8:1:1. For the mutualistic, gene regulatory, and neuronal dynamics, we consider 10, 5, and 11 different initial conditions, respectively. Box plots depict the median (central line) of F1-scores (n = 35 with different random seeds), the first and third quartiles (box), whiskers extending to 1.5 times the interquartile range from the first and third quartiles, respectively, and outliers are represented as individual points.

dynamics parameters and dynamics equations, which underscores the model's effectiveness in adapting to systems with varying node activity dynamics.

Networked systems with similar functions in the real-world often share identical forms of node activity dynamics, while the parameters governing these dynamics are usually different. For instance, the intrinsic mechanisms of self-growth and pairwise symbiotic effects on species abundances can be similarly described across a majority of ecosystems, however, the environmental capacities that support species vary significantly among them[7,39]. An intuitive idea is to leverage data from other systems with similar functions to enhance the resilience inference of the system under investigation, which necessitates that the inference model generalizes across various dynamics parameters. To assess such generalizability of ResInf, we generate node activities within networks using the same dynamics equations but with varying parameter settings (Supplementary Table 4), which are divided into training and test parameter settings with the ratio of 2:1. We train the models using networks configured with training parameter settings and subsequently evaluate them using those with unseen test parameter settings. Results for mutualistic, regulatory, and neuronal dynamics as depicted in Fig. 5a–c indicate that our model achieves the highest out-of-sample performance, with average F1-scores of 0.921, 0.892, and 0.924, respectively, demonstrating its superior generalizability. It reveals that ResInf effectively captures the common intrinsic dynamics form through the training data with different dynamics parameter settings.

Another crucial issue concerns whether ResInf can benefit from training on data of other systems, even those with different dynamic equations, and directly infer the resilience of investigated systems, which broadens the practical applications of our framework but also necessitates generalization across different dynamics equations. Previous works[7,40] have established that the resilience loss patterns of network systems governed by diverse dynamic equations can be classified into two main categories: phase shifts and the emergence of alternative stable states. For instance, Fig. 1g shows that network systems with mutualistic dynamics may undergo a transition from a region with a unique, non-zero equilibrium to another region with the emergence of multiple stable equilibria, where the system can be trapped in an undesired dysfunctional equilibrium. From Fig. 1h, gene regulatory networks may shift to a death equilibrium where all nodes have zero activity and can never recover to their functional equilibrium. Therefore, we hypothesize that if we train our ResInf framework on a dataset of diverse network systems covering both categories of resilience loss patterns, it should be able to generalize to network systems governed by novel dynamic equations that are previously unseen in the training set. To test this hypothesis, we introduce two new forms of node activity dynamics equations: susceptible-infectious-susceptible (SIS) dynamics[41] and inhibitory dynamics[42] (Dynamics and datasets in Methods). SIS and inhibitory dynamics belong to the categories of phase shifts and the emergence of alternative stable states, respectively. We construct the training set based on network systems governed by these two dynamics equations and test ResInf on the network systems governed by mutualistic, gene regulatory, and neuronal dynamics. As shown in Fig. 5d–f, we find that ResInf can achieve good performance and consistently outperforms all baseline models, even if it is tested on networks with new forms of node activity dynamics that have not been seen in the training set. It registers F1-scores of 0.862 ± 0.025, 0.889 ± 0.023, and 0.887 ± 0.019 on networked systems with mutualistic, regulatory, and neuronal dynamics, respectively, while other methods lose generalizability and have much worse performance. It is worth pointing out that analytical models (GBB and SDR) are not applicable in this generalization scenario because the dynamic equations of testing networks are unknown to the resilience inference model. These results suggest ResInf indeed has good generalizability, i.e., a representative training set is sufficient for ResInf to perform well in novel network systems with unknown node dynamic equations. We conduct further experiments where we train ResInf on networks with either SIS or inhibitory dynamics and evaluate its generalization performance on mutualistic, regulatory, and neuronal networks. We find that ResInf shows better generalizability when the resilience loss patterns of training networks cover those of the test samples (Supplementary Fig. 8a–c). It indicates that having network systems with the same category of resilience loss patterns in the training set ensures the generalization performance of ResInf. This requirement can be easily satisfied in various scenarios, and researchers can conveniently construct adequate training sets as we illustrated above.

Beyond assessing performance across various dynamics parameters and equations, we demonstrate that ResInf maintains its superior generalizability when tested on network topology types

https://doi.org/10.1038/s41467-024-53303-4

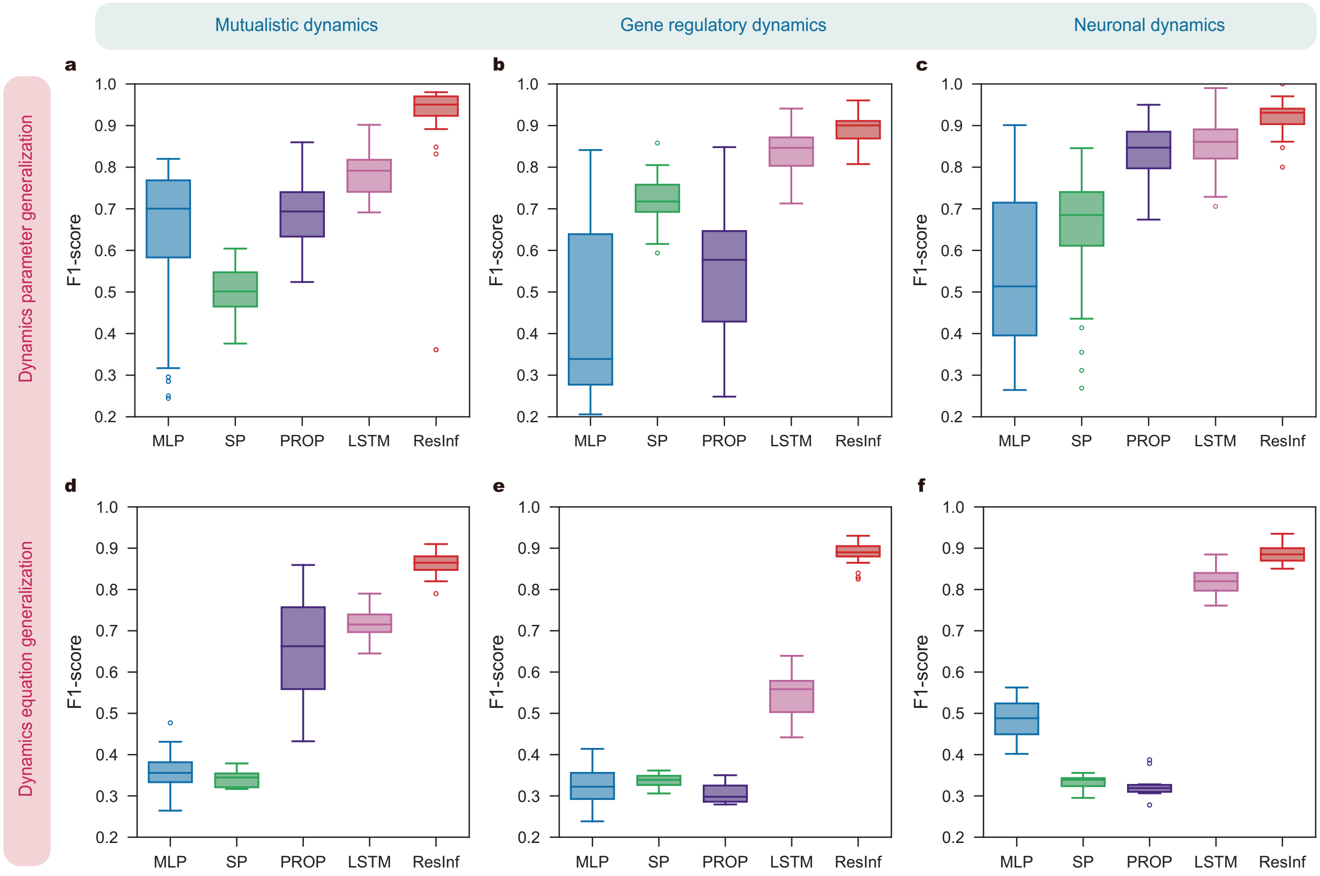

Fig. 5 | Generalizability of models. We demonstrate the generalizability of models across varying dynamics parameters and dynamics equations. **a**–**c** Generalizability of models across varying dynamics parameters of mutualistic (**a**), gene regulatory (**b**), and neuronal (**c**) dynamics. For each dynamics, we randomly choose six of nine parameter settings (Supplementary Table 4) for generating training data. We employ each parameter setting to simulate node activity trajectories for 1000 perturbed networks via node removal from empirical networks and combine all data of different parameter settings, formulating 6000 samples. To facilitate the training process, we randomly select 1000 samples to construct the training set. Similarly, we employ each of the other three unseen parameter settings to simulate node activity trajectories for 200 perturbed networks via node removal from empirical networks and combine all data of different parameter settings, formulating 600 samples and randomly selecting 200 samples to construct the validation/test set. **d**–**f** Generalizability of models across different dynamics equations. We employ three parameter settings of SIS and inhibitory dynamics (Supplementary Table 5) to simulate node activity trajectories for random networks with (N, p) uniformly sampled from [30,60) and [0.05,0.25), respectively. We synthesize 1600 samples under each parameter setting of each dynamics and combine all generated samples, then randomly select 1600 samples as the training set. To construct the validation/test set, we use mutualistic (**d**), gene regulatory (**e**), and neuronal (**f**) dynamics, respectively. We also employ three parameter settings of each dynamics (Index 2–4 of Supplementary Table 4) to simulate node activity trajectories for 160 random networks with the same (N, p) distributions as the training set. For mutualistic, gene regulatory, neuronal, SIS, and inhibitory dynamics, we consider 10, 5, 11, 11, and 11 different initial conditions, respectively. Box plots depict the median (central line) of F1-scores ($n = 35$ with different random seeds), the first and third quartiles (box), whiskers extending to 1.5 times the interquartile range from the first and third quartiles, respectively, and outliers are represented as individual points.

distinct from those used during training. After training on Erdős-Rényi networks with SIS and inhibitory dynamics, ResInf consistently obtains the best F1-scores among all baselines, surpassing over 0.818 on Barabási-Albert (scale-free) networks as well as empirical networks with mutualistic, regulatory, and neuronal dynamics, respectively (Supplementary Fig. 9). The GNN-based topology encoder of ResInf is capable of learning generalizable knowledge for resilience inference without making assumptions regarding topologies or relying on specific network structures, which significantly enhances the generalizability of ResInf to previously unseen topology types.

Resilience Inference from Noisy Data

Empirical measurements of real-world networked systems are often compromised by noise. Consequently, we evaluate the robustness of our model's performance in the presence of various types of observational noise, including noisy node activity trajectories, missing links, and spurious links. Specifically, we simulate noisy node activity trajectories by introducing Gaussian noise governed by a specified noise-to-signal ratio (NSR). Furthermore, missing and spurious links are simulated by randomly removing or adding a fraction of links to reproduce inaccurately estimated node interactions[43,44].

As depicted in Fig. 6a–c, it is evident that ResInf exhibits the best resilience inference performance when various levels of observational noise are introduced to the node activity trajectories. Taking the performance on mutualistic networks as an example, the F1-score of ResInf is 0.964 ± 0.011 in the absence of observational noise (NSR = 0), and it still remains at 0.854 ± 0.017 when NSR increases to 1, i.e., the power of the noise is as strong as the original node activity trajectories. The desirable noise-resistant feature of ResInf can be explained from the following aspects: first, the Transformer-based dynamics encoder can model temporal correlations and intrinsic dynamics of node activity from observed data and filter out uncorrelated noises adaptively[45–47]. The added noise is stochastic in nature, which will not easily blur the average, intrinsic patterns of system dynamics. Second, the topology encoder leverages graph neural network with a message-

https://doi.org/10.1038/s41467-024-53303-4

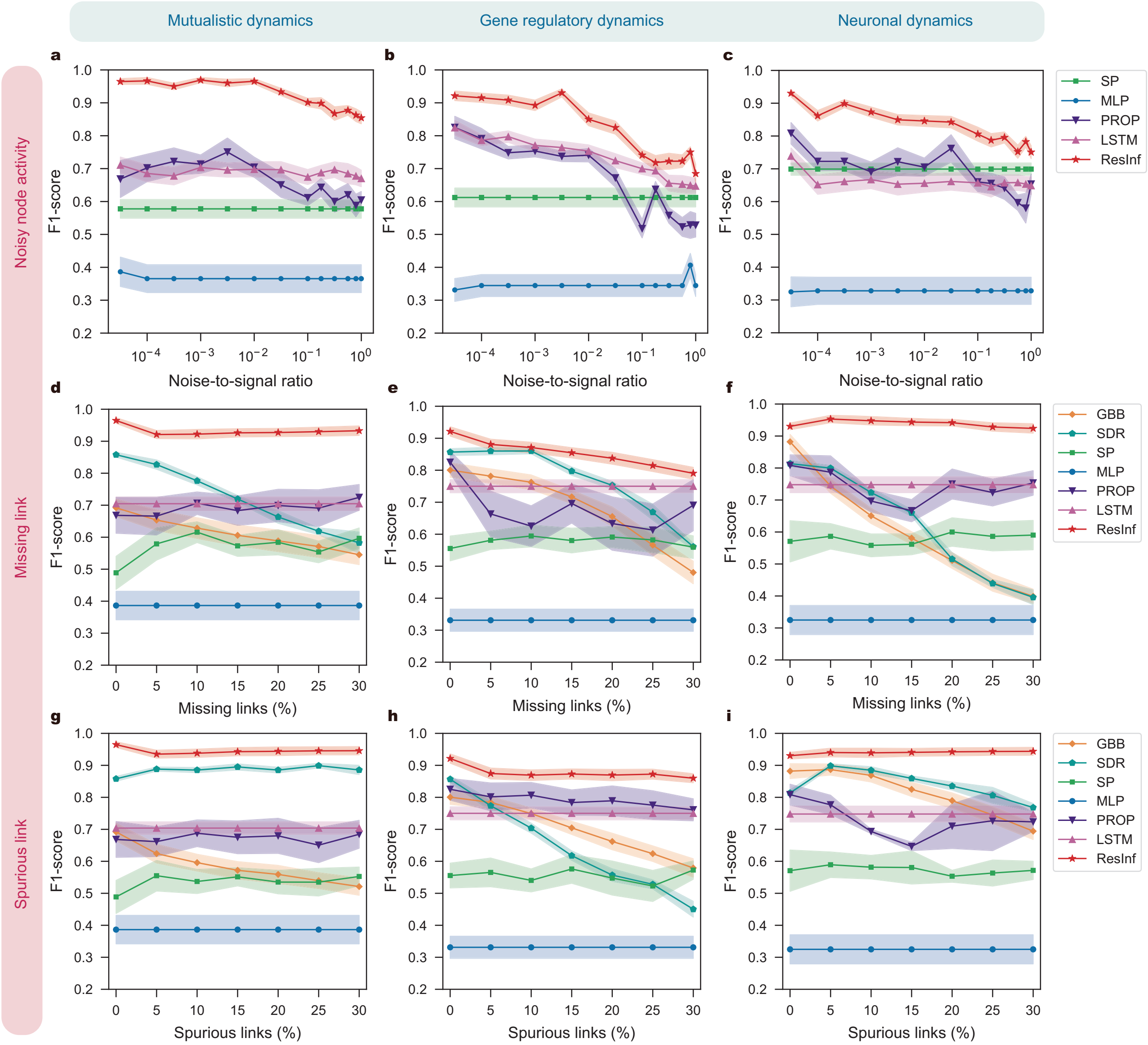

Fig. 6 | The robustness of resilience inference against observational noises. We employ the same datasets, data split ratios, the number of initial conditions considered, and dynamics parameter settings as in Fig. 4. In contrast, we introduce observational noises into their node activities or network topologies. We repeat all experiments with 35 different random seeds, corresponding to different inference model initialization and split for datasets. **a**–**c** Noisy node activity: we evaluate the models' performance when various intensity levels of Gaussian noises are added to the observed node activity trajectories. Higher noise-to-signal ratio indicates stronger noise. GBB and SDR are left out in this comparison because they are not affected by node activity trajectories. Instead, GBB and SDR rely on explicit network dynamic equations that are not available to other methods. **d**–**f** Missing link: we evaluate the models' performance on networked systems with 5 ~ 30% links randomly removed. **g**–**i** Spurious link: we randomly add 5 ~ 30% links to pairs of nodes that are not originally connected and evaluate the models' performance on the perturbed networked systems. Centers represent the average values, and shadings represent the standard deviation ($n = 35$ with different random seeds).

passing mechanism, which acts as a special form of Laplacian smoothing on the network, reducing the noise transmitted from neighboring nodes[26,48]. Therefore, the impact of noisy node activities can be significantly diminished, ensuring that the inference performance remains almost unaffected. It should be noted that analytical models GBB and SDR have access to precise governing dynamics equations and do not rely on observed activities. Consequently, they are unaffected by observational noise in node activity trajectories, making them not directly comparable in this context. Therefore, we assess the robustness of ResInf and analytical frameworks against the intrinsic uncertainty of the dynamics itself. Specifically, we introduce a stochastic term of Gaussian white noise with variance η into the ground truth dynamics equations to generate node activity trajectories. This uncertainty introduces a fundamental limitation in accurately describing the dynamics of node activities. From the results presented in Supplementary Fig. 10, we observe a significant decline or fluctuation in the performance of analytical frameworks, attributed to their reliance on estimating resilience solely by analyzing critical points from precise dynamics equations. Thus, these frameworks lack adaptability to real scenarios where dynamics are also influenced by intrinsic stochastic noises. In contrast, ResInf maintains remarkable performance at $\eta = 0.4$, exhibiting less susceptibility to the uncertainties of dynamic environments. These results further substantiate ResInf's superior capability in learning node activity dynamics and resilience patterns directly from observed data, showcasing a notable advantage over traditional frameworks.

https://doi.org/10.1038/s41467-024-53303-4

Regarding the noise in network topology, it has been found that ResInf can achieve a high F1-score of 0.933 ± 0.015 on mutualistic networks, even with 30% of the original links missing (Fig. 6d). Additionally, ResInf maintains a high F1-score of 0.945 ± 0.013 on mutualistic networks even when the rate of spurious links reaches 30% (Fig. 6g). ResInf displays similar performance curves across various levels of network topology noise in different settings (Fig. 6d–i). Conversely, the analytical frameworks GBB and SDR exhibit a notable decline in performance with increasing noise levels from missing and spurious links, highlighting their significant dependence on accurate network topologies. For example, on mutualistic networks, the F1-score of GBB significantly reduces from 0.691 ± 0.027 to 0.521 ± 0.026 as the percentage of missing links increases from 0% to 30%. The performances of MLP and LSTM remain unchanged with varying percentages of missing or spurious links, as they solely model the node activity trajectories. However, the significant performance gains of ResInf over MLP and LSTM underscore the importance of effectively representing network topologies. We further assess the models' robustness against the missing nodes, including cases where a fraction of nodes remains undetected either persistently or at a randomly chosen time step (Supplementary Fig. 11). These results further demonstrate that ResInf is better equipped to handle the effects of noisy network topologies and to ensure its potential efficacy in real-world applications. Given the correlated activities of real connected nodes, the dynamics encoder is capable of learning the activity dynamics of each node, accounting for influences from actual neighboring nodes. As resilience is inherently a network-level property, the network aggregator in the *k*-space projector (Fig. 2) adaptively aggregates information across the entire network. This approach effectively mitigates the impact of missing nodes or links, thereby enhancing the framework's robustness. In conclusion, these experimental results demonstrate that ResInf can still produce reliable inferences even in the presence of significant noise affecting both observed node activities and network topologies, highlighting a critical feature of ResInf that ensures its adaptability to real-world systems.

Visualization in 1-dimensional Decision Space

Beyond the empirical accuracy of resilience inference, the capability to visualize a system's proximity to the critical threshold of resilience loss within a low-dimensional decision space has profound implications for various applications. Such ability would allow stakeholders to make informed operation or management decisions in complex networked systems, which has been a useful feature of analytical approaches. In our study, ResInf uses *k*-space projector to map the networked systems into a condensed 1-dimensional *k*-space (Fig. 2), which facilitates accurate resilience inference with a simple monotonic classifier (ResInf framework in Methods). Therefore, the system representation in *k*-space can serve as an informative visualization of the system's resilience property. Experiments on previously mentioned synthetic networked systems show most resilient (red) and non-resilient (blue) systems are linearly separable in *k*-space (Fig. 7a–c). Specifically, by choosing a linear decision boundary with maximum likelihood k^c, we can classify the resilience systems with an F1-score of 0.933–0.987. The distance to the decision boundary in *k*-space indicates the system's proximity to the critical point of resilience loss. Therefore, *k*-space representations serve as an informative visualization tool and a reliable indicator of the system's resilience, offering a concrete way to interpret resilience inference results.

We compare the learned *k*-space with the β-space and α-space (Supplementary Note 1.6) generated by the analytical models GBB and SDR, respectively. We find the distributions of resilient and non-resilient systems have a larger overlap in β-space and α-space (Fig. 7d–i). For example, 47% of resilient mutualistic systems have β values smaller than the analytically inferred critical threshold β^c_{eff}, which will lead to inaccurate classification. As a result, the resilience inference enabled by β-space and α-space has significantly lower F1-score compared to ResInf. Additionally, we explore the ratio of true and false positives (inferred resilient systems) using the condensed 1-dimensional parameters and the decision spaces of GBB, SDR, and ResInf through the receiver operating characteristic (ROC) curve. The ROC curve indicates the trade-off between the true positive rate and the false positive rate as the discrimination threshold (ResInf framework in Methods and Supplementary Note 1.5–1.6) is varied. The area under the ROC curve (AUC) comprehensively quantifies both the sensitivity (the proportion of true positives correctly identified) and specificity (the proportion of false positives correctly avoided) of the model. An AUC of 1 indicates a perfect inference model, while an AUC of 0.5 signifies a model that performs no better than random chance (Evaluation metrics in Methods). We demonstrate ROC curves on the investigated dynamics systems (Fig. 7j–l) and find that AUCs of ResInf among these systems are all near to 1, outperforming the analytical framework. It shows that ResInf exhibits superior diagnostic accuracy. This performance underscores the effectiveness of ResInf in achieving a high true positive rate while maintaining a low false positive rate, demonstrating its superior capability of distinguishing between resilient and non-resilient networks and the reliable of employing deep learning approaches to resilience inference problems.

Discussion

In this paper, we propose a powerful deep learning framework, ResInf, to infer the resilience of complex networked systems solely based on the observational data of their topologies and node activities. It acts as a data-driven method that relieves the unrealistic assumptions of network topology and dynamics in previous state-of-the-art analytical approaches, e.g., the requirements of low assortativity topology, linear node activity dynamics, and knowing the explicit equations of node activity dynamics. The removal of these prerequisites largely broadens the application scenarios for more general complex networked systems, such as complex contagion in social networks[41,49], energy transmission in power grids[50,51], and predatory behavior in food webs[52], where the underlying dynamics often differs from case to case and cannot be known beforehand. Our experimental results demonstrate that ResInf exhibits strong generalizability across various networks with previously unseen dynamics of different equations and parameters. Particularly, by training on network data with the identified two main resilience loss patterns[7], ResInf consistently achieves superior performance on existing known node activity dynamics, such as mutualistic, gene regulatory, and neuronal dynamics, even when these dynamics are not present in the training set. Furthermore, ResInf demonstrates consistent generalizability across various network types, highlighting its ability to learn transferable knowledge for resilience inference. This underscores the model's applicability to a wide range of practical scenarios, particularly when encountering novel real-world network topologies and node activity dynamics. While the existing known node activity dynamics fall into the above two resilience loss patterns, future advances in this field may identify novel resilience loss patterns beyond these two categories, which could lead to sub-optimal performance of ResInf when such previously unseen patterns are emerging. Nevertheless, we can address this challenge by incorporating these novel patterns into the training set, thereby ensuring the generalizability of ResInf. Besides, as an analogy to the analytical β-space[5] and α-space[14], our ResInf model can map networked systems to a learned *k*-space where the resilient property of a given system can be more granularly gauged as the distance to critical threshold k^c, providing clear interpretable information on the system's current resilience status. As such, the system's k value serves as an insightful indicator that offers interpretability and fine-grained resilience assessment. Therefore, ResInf represents as a paradigm shift to data-driven resilience inference in empirical complex networked systems with interpretability. Beyond this, our recent work[38] paves the way to revive non-resilient networks. It suggests that non-resilient networks

https://doi.org/10.1038/s41467-024-53303-4

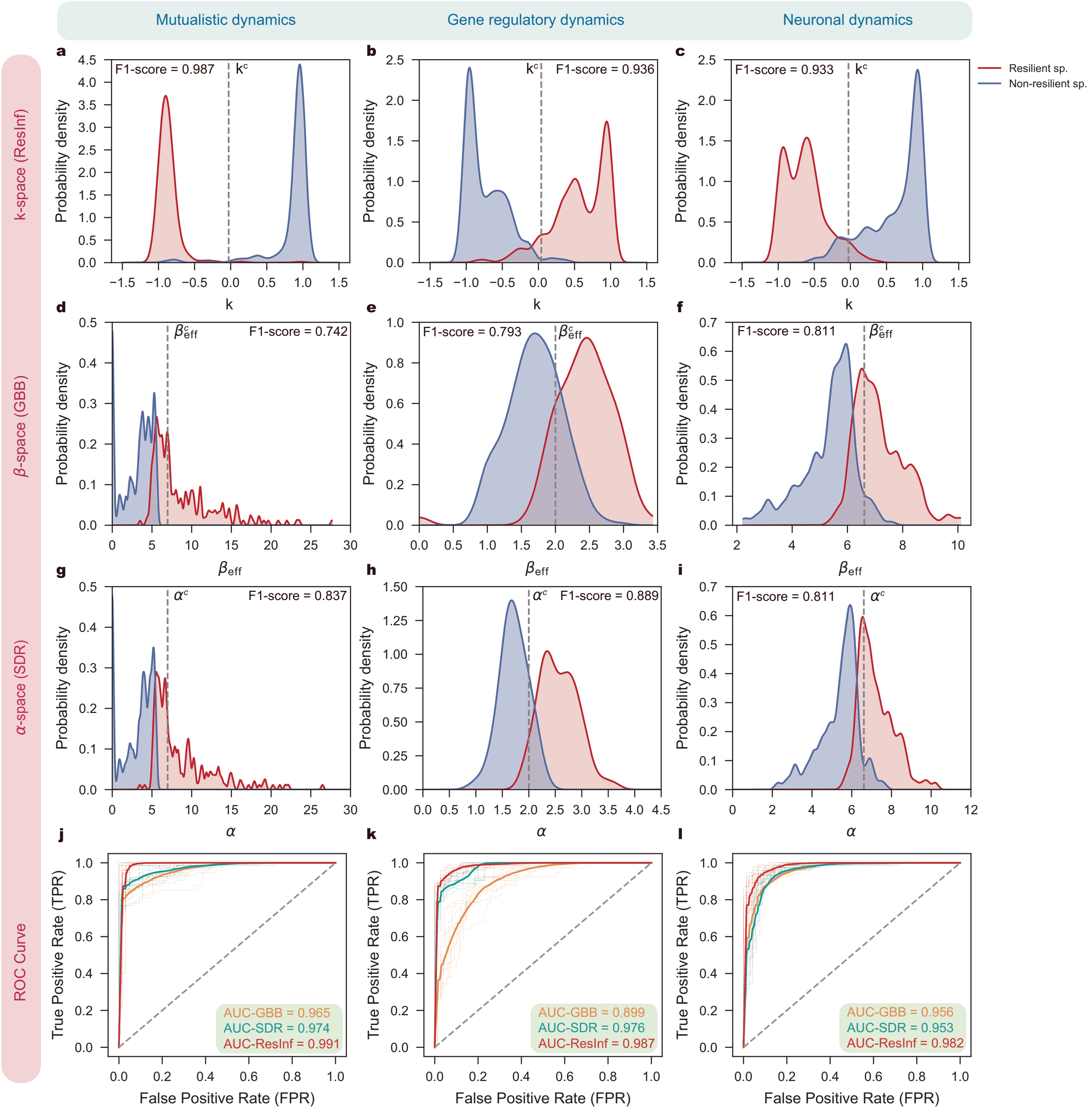

Fig. 7 | Visualizing networked systems in 1-dimensional decision space. We illustrate the probability distributions of k and β_{eff} and α for networked systems with mutualistic (**a, d, g**), gene regulatory (**b, e, h**), and neuronal (**c, f, i**) dynamics, taking one experiments of Fig. 4 for each dynamics as examples. The critical thresholds inferred by ResInf, k^c, are shown as the grey dashed lines in (**a**–**c**). The critical thresholds computed by GBB, β^c_{eff}, are shown as grey dashed lines in (**d**–**f**). We also show the critical threshold from SDR, α^c (**g**–**i**). We find resilient and non-resilient samples (resilient sp. and non-resilient sp.) are more linearly separable in k-space than β-space and α-space, which demonstrates the stronger resilience prediction capability of our proposed approach ResInf. **j**–**l** We illustrate the receiver operating characteristic (ROC) curve of GBB, SDR, and ResInf with the same experiment settings as Fig. 4 as well as the corresponding average AUC. Transparent lines represent experiments with different random seeds ($n = 50$), and solid lines represent the average ROC curve.

can be recovered to resilient ones by utilizing appropriate network restructuring and single-node reigniting strategies. ResInf can effectively guide such strategy design (Supplementary Note 5, Supplementary Fig. 14). Specifically, after implementing network restructuring and reigniting, we input the restructured network and initial time steps of node activities into ResInf. The mapped k point of the recovered network in the k-space acts as an indicator to ascertain the resilience recovery, which, in turn, enables us to refine and optimize the designed recovery strategy. This perspective shifts the focus of restoration from isolated interventions to a more integrated and systemic approach by prioritizing the network's structural integrity and the synergistic relationships among its components, which is essential for addressing complex real-world challenges.

To advance the depth and breadth of networked system resilience research, we suggest two potential directions for future investigation. First, despite the increasingly available observational data of complex networks, it remains a challenging task to sense high-quality node activity trajectories and topology in real-world networked systems,

https://doi.org/10.1038/s41467-024-53303-4

which often requires extensive laboratory environments[29]. As a result, it underscores the growing necessity for advanced complex network simulators. A promising research direction is to leverage generative AI frameworks to build learn-to-simulate models[49,53,54], which has the potential to go beyond the simple theoretical models[36,37] to generate more accurate simulation for wide-range of complex systems. Furthermore, while our proposed ResInf offers insights into the resilience inference challenge, the development of methods to enhance system resilience directly by AI methodologies is also promising for the research field. AI can be leveraged to build an optimization model that maximizes system resilience with a given budget, e.g., adding several critical links or dropping several weakly connected nodes. This would significantly amplify the impact of network resilience research as it extends to broader downstream applications[55] where enhancing resilience is crucial. To conclude, ResInf can serve as a proof-of-concept that exemplifies the potential of data-driven methods in network resilience research, demonstrating a promising avenue for innovating complex system studies with AI methods.

Methods

ResInf framework

Dynamics encoder. As previously mentioned, the underlying dynamics of networks mainly manifest themselves in node activities. Thus, to effectively capture node activity dynamics to infer their resilience, we develop a dynamics encoder inspired by Transformer[21,56] (Supplementary Fig. 1). We input the node activity trajectories $\mathbf{X} \in \mathbb{R}^{M \times N \times d}$, where d, M, N denote the used length of trajectories, the number of trajectories, and the number of nodes, respectively. For node i, its activity of the p-th trajectory can be denoted as $\mathbf{X}_i^p \in \mathbb{R}^{d \times 1}$. First, we transform activities of nodes at each time step with a shared multilayer perceptron (MLP), denoted as $\mathrm{f}_{\mathrm{in}} : \mathbb{R} \rightarrow \mathbb{R}^{d_e}$, that is, $\mathbf{e}_i^p = \mathrm{f}_{\mathrm{in}}(\mathbf{X}_i^p)$, where d_e denotes the embedding size at each time and $\mathbf{e}_i^p \in \mathbb{R}^{d \times d_e}$. We also add positional embeddings to the input embeddings to mark the relative positions, i.e., $\mathbf{e}_i^p \leftarrow \mathbf{e}_i^p + \mathrm{PE}$. At each time step t of the total d time steps, positional embeddings are formulated as $\mathrm{PE}_{(t,2n)} = \sin(t/10000^{2n/d_e})$, $\mathrm{PE}_{(t,2n+1)} = \cos(t/10000^{2n/d_e})$, where the second element in the tuple of the subscript ($2n$ and $2n + 1$) corresponds to the index of elements in positional embeddings. Each dynamics encoder layer comprises a self-attention sub-layer and a fully-connected feed-forward sub-layer. A residual connection is established between these two sub-layers, which is subsequently followed by the application of layer normalization[57]. Taking the input $\mathbf{e}$ as an example, the output of the sub-layer can be formulated as $\mathrm{LayerNorm}(\mathbf{e} + \mathrm{Sublayer}(\mathbf{e}))$.

The self-attention sub-layer consists of multiple attention heads, which enable ResInf to concurrently integrate information from different representation subspaces while simultaneously account for diverse positional contexts. For the attention head j, taking the input $\mathbf{e}_i^p$ as an example, the result embeddings can be obtained from the query matrix $\mathbf{Q}_j$ and key-value pairs $\mathbf{K}_j$ and $\mathbf{V}_j$, which can be formulated as:

$$\mathrm{head}_j = \mathrm{softmax}\left(\frac{\mathbf{Q}_j \mathbf{K}_j^T}{\sqrt{d_k}}\right)\mathbf{V}_j, \tag{2}$$

$$\mathbf{Q}_j = \mathbf{e}_i^p \mathbf{W}_j^Q, \mathbf{K}_j = \mathbf{e}_i^p \mathbf{W}_j^K, \mathbf{V}_j = \mathbf{e}_i^p \mathbf{W}_j^V, \tag{3}$$

where $\mathbf{W}_j^Q \in \mathbb{R}^{d_e \times d_k}$, $\mathbf{W}_j^K \in \mathbb{R}^{d_e \times d_k}$, and $\mathbf{W}_j^V \in \mathbb{R}^{d_e \times d_e}$ are trainable parameters. Fusing the information from h attention heads, we can derive the multi-head encoded representation, which also serves as the output of the self-attention sub-layer, formulated as:

$$\mathrm{Self-attention}(\mathbf{e}_\mathrm{i}^\mathrm{p}) = \mathrm{Concat}(\mathrm{head}_1, \mathrm{head}_2, \ldots, \mathrm{head}_\mathrm{h})\mathbf{W}^\mathrm{O}, \tag{4}$$

where $\mathbf{W}^O \in \mathbb{R}^{h \cdot d_e \times d_e}$ are trainable parameters. Therefore, the final output of the self-attention sub-layer can be formulated as:

$$\tilde{\mathbf{e}}_i^p = \mathrm{LayerNorm}(\mathbf{e}_i^p + \mathrm{Self-attention}(\mathbf{e}_\mathrm{i}^\mathrm{p})), \tag{5}$$

where $\tilde{\mathbf{e}}_i^p \in \mathbb{R}^{d \times d_e}$ represents the final output from the self-attention sub-layer.

The fully-connected feed-forward sub-layer comprises two linear transformation layers, with a ReLU activation function applied in between them. The output of the feed-forward sub-layer can be formulated as:

$$\mathrm{Feed-forward}(\mathbf{e}_\mathrm{i}^\mathrm{p}) = \mathrm{ReLU}(\mathbf{e}_\mathrm{i}^\mathrm{p}\mathbf{W}_1 + \mathbf{b}_1)\mathbf{W}_2 + \mathbf{b}_2, \tag{6}$$

where $\mathbf{W}_1 \in \mathbb{R}^{d_e \times d_h}$, $\mathbf{b}_1 \in \mathbb{R}^{d_h}$, $\mathbf{W}_2 \in \mathbb{R}^{d_h \times d_e}$ and $\mathbf{b}_2 \in \mathbb{R}^{d_e}$ are trainable parameters. Therefore, the final output of the fully-connected feed-forward sub-layer can be formulated as:

$$\hat{\mathbf{e}}_i^p = \mathrm{LayerNorm}(\tilde{\mathbf{e}}_i^p + \mathrm{Feed-forward}(\mathbf{e}_\mathrm{i}^\mathrm{p})), \tag{7}$$

where $\hat{\mathbf{e}}_i^p \in \mathbb{R}^{d \times d_e}$ represents the final output from the fully-connected feed-forward sub-layer.

During the encoding process, $\mathbf{e}_i^p$ serves as the input to the first layer of the dynamics encoder, yielding the output $\hat{\mathbf{e}}_i^p$. Subsequently, $\hat{\mathbf{e}}_i^p$ becomes the input for the next layer of the dynamics encoder if we use multiple layers of the dynamics encoder. The output from the final layer of the dynamics encoder, denoted as $\mathbf{Z}_p^i \in \mathbb{R}^{d \times d_e}$, plays the role of the node activity dynamics representation for node i of the p-th trajectory. It should be noted that all nodes and trajectories, i.e., $(i, p) \in \{1, 2, \cdots, |V|\} \times \{1, 2, \cdots, M\}$, are encoded via the dynamics encoder in a parallel manner. We integrate the output representations of the last time step for nodes as their final node activity dynamics representations of the p-th trajectory, which is denoted as $\mathbf{Z}_p \in \mathbb{R}^{N \times d_e}$.

Topology encoder. We design a graph neural network (GNN)[22] with multi-hop message passing to model the impact from multi-hop neighborhoods of nodes (Supplementary Fig. 2). Given an adjacency matrix $\mathbf{A}$, we design the message passing operator $\mathbf{\Psi}$ as $\mathbf{\Psi} = \mathbf{I}_N - \mathbf{D}_{\mathrm{in}}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}_{\mathrm{out}}^{-\frac{1}{2}}$, where the diagonal element in the i-th row of $\mathbf{D}_{\mathrm{in}}$ and $\mathbf{D}_{\mathrm{out}}$ denotes the in-degree and out-degree of node i, respectively. The l-th layer message passing can be formulated as follows:

$$\mathbf{Z}_p^{(l)} = \tanh(\mathbf{Z}_p^{(l-1)}\mathbf{W}_F^{(l)}) + \tanh(\mathbf{\Psi}\mathbf{Z}_p^{(l-1)}\mathbf{W}_G^{(l)}), \tag{8}$$

where $\mathbf{Z}_p^{(l)}$ represents node representations after l layers of message passing. $\mathbf{W}_F^{(l)} \in \mathbb{R}^{d_e \times d_e}$ and $\mathbf{W}_G^{(l)} \in \mathbb{R}^{d_e \times d_e}$ are trainable transformation matrices of the l-th layer, respectively. Since resilience is a system-level property, we introduce a virtual node to fuse the messages from all individual nodes called global pooling. With the virtual global node, the expanded node representation matrix of the p-th trajectory after the l-th message passing layer is denoted as $\mathbf{Z}_p^{*(l)} \in \mathbb{R}^{(N+1) \times d_e}$.

***k*-space projector**. We introduce an attention module that can effectively integrate the embeddings of M trajectories (Supplementary Fig. 3). Inspired by the existing work[58], we design pooling layers that can adaptively capture crucial information for each of the M trajectories. Specifically, we calculate the average and max pooling of $\mathbf{Z}_p^{*(l)}$, denoted as $\mathbf{Z}_{\mathrm{avg}}^{(l)} \in \mathbb{R}^{M \times 1 \times 1}$ and $\mathbf{Z}_{\mathrm{max}}^{(l)} \in \mathbb{R}^{M \times 1 \times 1}$, respectively. Then the attention weight of trajectories can be calculated as:

$$\mathbf{Att} = \sigma(\mathrm{MLP}(\mathbf{Z}_{\mathrm{avg}}^{(l)}) + \mathrm{MLP}(\mathbf{Z}_{\mathrm{max}}^{(l)})), \tag{9}$$

where $\mathbf{Att} \in \mathbb{R}^{M \times 1 \times 1}$ and $\sigma(x) = \frac{1}{1+e^{-x}}$ is the sigmoid activation function.

https://doi.org/10.1038/s41467-024-53303-4

After that, we fuse representations from all M trajectories with the attention weight, formulated as

$$\mathbf{Z}=\sum_{p=1}^{M}\mathrm{Att}_p * \mathbf{Z}_p^{*(l)}, \tag{10}$$

where $\mathrm{Att}_p \in \mathbb{R}$ denotes the attention weight of the p-th trajectory and $\mathbf{Z} \in \mathbb{R}^{(N+1)\times d_e}$ is the final representation of all nodes. We derive the network representation by employing the final representation of the virtual node, denoted as $\mathbf{e}_G$. Subsequently, we propose an MLP-based resilience predictor that leverages $\mathbf{e}_G \in \mathbb{R}^{d_e}$ to infer the resilience of the network (Supplementary Fig. 3). In this predictor, we first map the network to a scalar k of 1-dimensional k-space and then provide the inferred network resilience $\hat{y}$ as follows:

$$k=\mathrm{MLP}(\mathbf{e}_G), \tag{11}$$

$$\hat{y}=H(\sigma(wk+b)-\gamma), \tag{12}$$

where $w, b \in \mathbb{R}$ are the trainable weight and bias, respectively. $\sigma(\cdot)$ denotes the sigmoid activation function, i.e., $\sigma(x)=\frac{1}{1+e^{-x}}$, and $H(\cdot)$ represents the Heaviside step function. Without loss of generality, we set γ = 0.5 as the resilience inference threshold. Since the function $f(k)=\sigma(wk+b)-\delta$ is monotonic, there is only one zero point, k^c, which is defined as the critical threshold in k-space. $k<k^c$ and $k>k^c$ lead to the different prediction $\hat{y}$, indicating whether the network is resilient ($\hat{y}=1$) or not ($\hat{y}=0$).

The loss function used to train ResInf consists of two parts: binary cross-entropy term and l_2-regularization term, which is described as follows:

$$L=-\frac{1}{S}\sum_{i=1}^{S}y_i\log(\hat{y}_i)+(1-y_i)\log(1-\hat{y}_i)+\lambda||\Theta||_2, \tag{13}$$

where L is the loss function we aim to minimize. $\hat{y}_i$ denotes the resilience inference result of ResInf for the i-th sample, and y_i represents its resilience i.e., label. S denotes the number of training samples. Θ denotes all trainable parameters in ResInf and λ is a hyperparameter to control the relative weight of l_2-regularization term. We use the Adam optimizer[59] to minimize the loss function during training.

Real-world microbial systems

The dataset is comprised of species abundances of different microbial communities observed during 11 days of experiments. Each microbial community belongs to one of the species pool sizes: 3 species, 6 species, 12 species, and 24 species, and is cultivated under one of the high, medium, or low nutritional conditions. Previous research[29] demonstrates that high nutritional conditions and larger species pool sizes foster intense competition among species, driving the system to a non-resilient phase featured by chaotic abundance fluctuations, as opposed to a resilient phase marked by stable abundance equilibrium[30]. To differentiate between resilient and non-resilient communities and determine their resilience labels, consistent with the existing work[29], we calculate the average coefficient of variation ($\overline{\mathrm{CV}}$) of observed species abundance of Day 7 ~ 10 in the community. It corresponds to the average value of the standard deviation for the abundance of each species over these days, scaled by the average species abundance of these days, which can be represented as follows:

$$\overline{\mathrm{CV}}=\frac{\langle \text{Std of } x_i(t)\rangle}{\langle \text{Mean of } x_i(t)\rangle}, \tag{14}$$

where t runs from 7 to 10, and $x_i(t)$ is the i-th element of $\mathbf{x}(t)$ at the time t. Systems with $\overline{\mathrm{CV}}$ in this time interval lower than a threshold 0.25 are considered converged to a stable equilibrium as resilient systems[29]. If $\overline{\mathrm{CV}}$ is larger than the threshold, the investigated systems are non-resilient. A high $\overline{\mathrm{CV}}$ indicates that the system lacks a desired stable equilibrium and is instead trapped in persistent fluctuations.

Simulating critical events

With the synthetic and empirical networks, we simulate critical events that may result in resilience loss by employing perturbations according to the experimental setup outlined in previous study[5]. These perturbations include node removal, link removal, and global weight perturbation, which are described as follows.

Node removal. To model node failure in networked systems, we randomly select and remove n nodes ($n<N$) and their associated links. After the deletion, we reconstruct the derived giant component of the network and eliminate the isolated nodes.

Link removal. To model the critical events on interactions between nodes, we randomly select and remove m links, where m is less than the total number of links $|E|$. We also eliminate the isolated nodes from the derived giant component.

Global weight perturbation. To model the impact of environmental changes on a global scale, we introduce a macroscopic perturbation that affects all link weights in the network. Specifically, we apply a random factor, denoted as r_{ij}, to shift the weight of the edge between node i and j such that A_{ij} is replaced with $r_{ij}A_{ij}$. The r_{ij} value is drawn from a uniform distribution with a mean of $r<1$. As a result, the network is transformed into a perturbed state where the interaction weights are reduced to a fraction r of their original value.

Evaluation metrics

F1-score. The primary metric employed to evaluate the model's inference ability is the F1 score. The F1-score for positive samples can be formulated as:

$$\mathrm{F1-score}=2\frac{\mathrm{precision}\cdot\mathrm{recall}}{\mathrm{precision}+\mathrm{recall}}=2\frac{\mathrm{TruePositive}}{2\cdot\mathrm{TruePositive}+\mathrm{FalsePositive}+\mathrm{FalseNegative}}, \tag{15}$$

$$\mathrm{TruePositive(TP)}=\sum_{i=1}^{P}1\left(y_i=1\wedge\hat{y}_i=1\right), \tag{16}$$

$$\mathrm{FalsePositive(FP)}=\sum_{i=1}^{P}1\left(y_i=0\wedge\hat{y}_i=1\right), \tag{17}$$

$$\mathrm{TrueNegative(TN)}=\sum_{i=1}^{P}1\left(y_i=0\wedge\hat{y}_i=0\right), \tag{18}$$

$$\mathrm{FalseNegative(FN)}=\sum_{i=1}^{P}1\left(y_i=1\wedge\hat{y}_i=0\right), \tag{19}$$

$$\mathrm{precision}=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}, \tag{20}$$

$$\mathrm{recall}=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}, \tag{21}$$

where P denotes the number of all test samples, y_i is the resilience label of the i-th system, and $\hat{y}_i$ denotes the resilience inference result from the model. We can similarly calculate the F1-score for negative samples by interchanging the roles of "positive" and "negative" in the computation process of Equation (15). Given the potential imbalance between positive and negative samples in test datasets, we report the

https://doi.org/10.1038/s41467-024-53303-4

weighted average of F1-score for positive samples and that for negative samples, where the weights are determined by the number of instances in each class.

Receiver operating characteristic curve (ROC). ROC curve demonstrates the performance of inference models with varying discrimination threshold γ in Equation (12), illustrating the true positive rate (TPR) with respect to the false positive rate (FPR). For example, if we lower the threshold γ in Equation (12), more samples will be regarded as resilient networks ($\hat{y}_i = 1$), thus increasing the number of false positive as well as true positive samples and corresponding to a different point on the curve. TPR and FPR can be formulated as follows:

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \tag{22}$$

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}}, \tag{23}$$

where TP, FP, TN, FN are defined in Equation (16–19).

Area under the ROC curve (AUC). Area Under the ROC Curve (AUC) quantifies the overall ability of the model to discriminate between classes across all possible thresholds γ in Equation (12), offering a comprehensive assessment of the model's performance. Essentially, a higher AUC value for the model indicates that it is more likely to rank a randomly selected positive instance higher than a randomly chosen negative one, i.e., higher value of $\sigma(\mathrm{w}k + b)$ in Equation (12), for this positive sample. An AUC of 1.0 indicates a perfect inference model with flawless predictive capability, while an AUC of 0.5 signifies a model performing no better than random guessing.

Dynamics and datasets

Mutualistic dynamics. The mutualistic dynamics can be formulated as:

$$\frac{dx_i}{dt} = B + x_i\left(1 - \frac{x_i}{K}\right)\left(\frac{x_i}{C} - 1\right) + \sum_{j=1}^{N} A_{ij} \frac{x_i x_j}{D + Ex_i + Hx_j}. \tag{24}$$

where B denotes the incoming rate for species i from neighbor ecosystems. K indicates the environment carrying capacity, and C expresses the Allee effect, implying when $x_i < C$ there is a negative effect for node activities. The third term describes symbiotic interactions with the saturated effect for the positive contribution of node j to the activity of node i. We illustrate the dynamics parameter settings for experiments in Supplementary Table 4. Mutualistic dynamics can encounter a bifurcation, where the system undergoes a transition from the resilient phase with only a single desired stable equilibrium $\mathbf{x}^H$ to the non-resilient phase where both desired equilibrium $\mathbf{x}^H$ and undesired equilibrium $\mathbf{x}^L$ are stable, indicating that non-resilient networks can collapse to undesired equilibrium $\mathbf{x}^L$ in the face of massive perturbations or low initial conditions of node activities.

We retrieve empirical networked systems from the existing works[5,60] and the Interaction Web Database (IWDB)[35], which contains mutualistic systems such as plant-ant, plant-pollinator, and anemone-fish. The detailed information is shown in Supplementary Table 1. Taking the mutualistic system of plant-ant as an example, the system consists of a plants and b ants, forming an bipartite network $\mathbf{B}$ with the size of $a \times b$. Then we project the bipartite network to plant/ant set, resulting in two networks which contain a plants ($\mathbf{B}^{\mathrm{I}}$) and b ants ($\mathbf{B}^{\mathrm{II}}$), respectively. Link weights B^{I}_{ij} in the projection network of plant between node i and j can be formulated as $B^{\mathrm{I}}_{ij} = \sum_{k=1}^{b} \frac{B_{ik}B_{jk}}{\sum_{s=1}^{a} B_{sk}}$. The formula indicates that the symbiotic interaction between plant i and j depends on two aspects: (i) the number of ants with which both plant i and j have interactions; (ii) the number of plants with which the ant k has interactions. The increase of the number in (i) will strengthen the weight B^{I}_{ij}, while the increase of the number in (ii) will have the opposite impact. We conduct a similar procedure to the projection network of ant $\mathbf{B}^{\mathrm{II}}$ to calculate its link weight B^{II}_{ij}. After the network projection, there may be isolated nodes whose activities no longer rely on the primary structure of the network. Hence, we remove these isolated nodes and reconstruct the giant component of the network, which is a connected sub-network containing the maximum proportion of the original network's nodes. To model the effect of critical events that can lead to resilience loss, we apply perturbations of different intensities, including the random removal of nodes and links, and global perturbation on link weights, to giant components of networks (Simulating critical events in Methods). The statistics of datasets comprising the derived networks are illustrated in Supplementary Table 2. We then obtain their node activity trajectories by simulating differential equation groups (24) of mutualistic dynamics using the fourth-order Runge-Kutta stepper[61] with M different initial activities sampled from a logarithmic scale $[10^{-2}, 10^{1})$ for each node. The terminal simulation time is $T_{\max} = 200$. The ground truths of network resilience are determined by node activities at $T_{\max}$. We obtain the average terminal activity $\langle \mathbf{x}_{\mathrm{end}} \rangle_m$ of nodes for each of M trajectories, respectively, and calculate the sum of distances between each average activity $\langle \mathbf{x}_{\mathrm{end}} \rangle_m$ and their mean value $\overline{\langle \mathbf{x}_{\mathrm{end}} \rangle}$, which is denoted as D_1, i.e., $D_1 = \sum_{m=1}^{M} \left\| \langle \mathbf{x}_{\mathrm{end}} \rangle_m - \overline{\langle \mathbf{x}_{\mathrm{end}} \rangle} \right\|_2$. In addition, we use K-means to cluster these M average activities into two classes and calculate the sum of distances between each average activity $\langle \mathbf{x}_{\mathrm{end}} \rangle_m$ and its cluster center $\mathbf{x}^{\mathrm{center}}_m$, which is denoted as D_2, i.e., $D_2 = \sum_{m=1}^{M} \left\| \langle \mathbf{x}_{\mathrm{end}} \rangle_m - \mathbf{x}^{\mathrm{center}}_m \right\|_2$. If $D_2 < \frac{D_1}{10}$, we think the clusters generated from K-means are more reliable and regard the network as non-resilient, with two stable equilibria $\mathbf{x}^H$ and $\mathbf{x}^L$.

Gene regulatory dynamics. The gene regulatory dynamics can be formulated as:

$$\frac{dx_i}{dt} = -Bx_i^f + \sum_{i=1}^{N} A_{ij} \frac{x_j^h}{x_j^h + 1}, \tag{25}$$

which is also called Michaelis-Menten dynamics[3]. f can represent degradation (f = 1) or dimerization (f = 2), and the second term describes genetic activation, where h is the Hill coefficient and explains the degree of collaboration in gene regulation. We illustrate the dynamics parameter settings for experiments in Supplementary Table 4. $\mathbf{x}^L = \mathbf{0}$ (also $\langle x \rangle = 0$) is the trivial and undesired stable equilibrium. Resilient systems have another desired equilibrium $\mathbf{x}^H$ where the average node activity $\langle x \rangle > 0$. Non-resilient systems cannot stay in $\mathbf{x}^H$, indicating cell death with only zero node activity ($\langle x \rangle = 0$).

We collect empirical regulatory networks from yeast[62] and *E.coli*[63], whose detailed information is shown in Supplementary Table 1. Since these networks are directed, we focus on their strongly connected components, where there is a path in both directions between each pair of nodes, and adopt the same kinds of perturbations as mutualistic networks. The statistics of datasets comprising the derived networks are illustrated in Supplementary Table 2. Node activity trajectories of these networks are obtained from differential equation groups (25) of gene regulatory dynamics using the fourth-order Runge-Kutta stepper[61] with M different initial activities sampled from [3, 10) for each node, and the terminal simulation time is $T_{\max} = 200$. The ground truths of network resilience are determined by node activities at $T_{\max}$. We obtain the average terminal activity of nodes for M trajectories and further calculate their mean value $\overline{\langle x \rangle}$. If $\overline{\langle x \rangle} < 10^{-5}$, we can approximate $\overline{\langle x \rangle} = 0$ and the network will be non-resilient.

https://doi.org/10.1038/s41467-024-53303-4

Neuronal dynamics. The neuronal dynamics can be formulated as

$$\frac{dx_i}{dt} = -x_i + \sum_{j=1}^{N} A_{ij}\frac{1}{1+e^{\mu-\delta x_j}}, \tag{26}$$

which is also called Wilson-Cowan dynamics[20,64]. Since each node receives cumulative inputs from all of its neighbors, higher in-degree nodes benefit from their surroundings by having a larger aggregate activation signal. We illustrate the dynamics parameter settings for experiments in Supplementary Table 4. Non-resilient neuronal networks exhibit either a bi-stable phase with both stable desired and undesired equilibria $\mathbf{x}^H$ and $\mathbf{x}^L$, respectively, or have only a single undesired stable equilibrium $\mathbf{x}^L$. Resilient neuronal networks only have a single desired equilibrium $\mathbf{x}^H$.

Following existing researches[38], we use Erdős-Rényi[36] and Barabási-Albert[37] models to create neuronal networks, whose detailed information is shown in Supplementary Table 1. We obtain their giant components and adopt the same kinds of aforementioned perturbations. The statistics of datasets comprising the derived networks are illustrated in Supplementary Table 2. Node activity series of these networks are simulated with differential equation groups (26) of neuronal dynamics using the fourth-order Runge-Kutta stepper[61] with M different initial activities sampled from $[10^{-2}, 10^{1})$ for each node, and the terminal simulation time is $T_{\max} = 200$. The ground truths of network resilience are determined by node activities at $T_{\max}$. First, via the same approach as in mutualistic networks, we calculate and compare D_1 and D_2 to find networks with two stable equilibria $\mathbf{x}^H$ and $\mathbf{x}^L$, and regard them as non-resilient. For the remaining networks, we regard those whose unique stable equilibrium is closer to $\mathbf{x}^H$ than $\mathbf{x}^L$ as resilient networks. Otherwise, we consider them as non-resilient networks.

SIS dynamics. The susceptible-infected-susceptible (SIS) dynamics[41] can be formulated as:

$$\frac{dx_i}{dt} = -\delta x_i + \sum_{j=1}^{N} A_{ij}(1-x_i)x_j, \tag{27}$$

which describes the spreading phenomena of the epidemic. The node activity x_i represents the infection probability of node i and A_{ij} denotes the infection rate from node j to node i. $\delta > 0$ denotes the curing rate of node i. The parameter settings are detailed in Supplementary Table 5. Without loss of generality, we define SIS networks with equilibrium $\mathbf{x}^L = \mathbf{0}$ (also $\langle x\rangle = 0$) as non-resilient networks, while those with equilibrium $\mathbf{x}^H > \mathbf{0}$ (also $\langle x\rangle > 0$) are resilient networks. We employ equation groups (27) with the fourth-order Runge-Kutta stepper to simulate node activity trajectories for networks with M different initial activities sampled from a uniform distribution $[0, 1)$ for each node, and the terminal simulation time is $T_{\max} = 8.5$. The ground truths of network resilience are determined by node activities at $T_{\max}$. We obtain the average terminal activity of nodes for M trajectories and further calculate their mean value $\overline{\langle x\rangle}$. If $\overline{\langle x\rangle} < 10^{-5}$, we can approximate $\overline{\langle x\rangle} = 0$ and the network will be non-resilient.

Inhibitory dynamics. The inhibitory dynamics[42] can be formulated as:

$$\frac{dx_i}{dt} = c + x_i\left(1-\frac{x_i}{f}\right)\left(\frac{x_i}{b}-1\right) + \sum_{j=1}^{N} A_{ij}\frac{x_i}{1+x_j}, \tag{28}$$

which models inhibition between genes[65] or between hosts and pathogens[42]. The self dynamics describes the logistic growth with the Allee effect, and the interaction dynamics capture the inhibition, where the population of i grows linearly with x_i, but when $x_j \to \infty$, the rate approaches zero. Therefore, the population of j will inhibit the growth of i. The parameter settings are detailed in Supplementary Table 5. Inhibitory dynamics encounter a bifurcation, where the system transits from the resilient regime with a desired unique equilibrium $\mathbf{x}^H$ to the non-resilient regime with both the desired equilibrium $\mathbf{x}^H$ and the undesired equilibrium $\mathbf{x}^L$. Consequently, the system can collapse to $\mathbf{x}^L$ in the face of massive perturbations or low initial conditions of node activities. We employ equations group (28) with the fourth-order Runge-Kutta stepper to simulate node trajectories for networks with M initial activities sampled from a uniform distribution $[0, 1]$ for each node, and the terminal simulation time is $T_{\max} = 20$. The ground truths of network resilience are determined by node activities at the terminal time $T_{\max}$. Similar to the process of mutualistic dynamics, we calculate D_1 and D_2 for each network and regard networks with $D_2 < \frac{D_1}{10}$ as non-resilient networks.

Reporting summary
Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability
The data of topologies and node activity trajectories used in this study are available and can be freely accessed through GitHub (https://github.com/tsinghua-fib-lab/ResInf)[66]. The empirical network topology data is also available at the Interaction Web Database (http://www.ecologia.ib.usp.br/iwdb/resources.html), Copenhagen Network Study (https://doi.org/10.6084/m9.figshare.7267433), SocioPatterns Project (http://www.sociopatterns.org/datasets), and https://github.com/jianxigao/NuRsE. The laboratory microbial systems data is also available at https://www.science.org/doi/10.1126/science.abm7841. Source data are provided with this paper.

Code availability
All codes used for data generation, simulation, and analysis in this research are deposited in GitHub (https://github.com/tsinghua-fib-lab/ResInf) and Zenodo[66].

References

1. May, R. M. Thresholds and breakpoints in ecosystems with a multiplicity of stable states. *Nature* **269**, 471–477 (1977).
2. Barabási, A.-L. & Albert, R. Emergence of scaling in random networks. *Science* **286**, 509–512 (1999).
3. Alon, U. *An Introduction to Systems Biology: Design Principles of Biological Circuits*. (Chapman and Hall/CRC, London, 2006).
4. Barabási, A.-L. & Pósfai, M. *Network Science*. (Cambridge University Press, Cambridge, 2016).
5. Gao, J., Barzel, B. & Barabási, A.-L. Universal resilience patterns in complex networks. *Nature* **530**, 307–312 (2016).
6. Cohen, R., Erez, K., Ben-Avraham, D. & Havlin, S. Resilience of the internet to random breakdowns. *Phys. Rev. Lett.* **85**, 4626 (2000).
7. Liu, X. et al. Network resilience. *Phys. Rep.* **971**, 1–108 (2022).
8. May, R. M. Will a large complex system be stable? *Nature* **238**, 413–414 (1972).
9. Holling, C. S. Resilience and stability of ecological systems. *Annu. Rev. Ecol. Syst.* **4**, 1–23 (1973).
10. Albert, R., Jeong, H. & Barabási, A.-L. Error and attack tolerance of complex networks. *Nature* **406**, 378–382 (2000).
11. Lyapunov, A. M. The general problem of the stability of motion. *Int. J. Control* **55**, 531–534 (1992).
12. Barzel, B. & Biham, O. Quantifying the connectivity of a network: The network correlation function method. *Phys. Rev. E* **80**, 046104 (2009).
13. Zylstra, E. R. et al. Changes in climate drive recent monarch butterfly dynamics. *Nat. Ecol. Evolution* **5**, 1441–1452 (2021).
14. Laurence, E., Doyon, N., Dubé, L. J. & Desrosiers, P. Spectral dimension reduction of complex dynamical networks. *Phys. Rev. X* **9**, 011042 (2019).

15. Jiang, C., Gao, J. & Magdon-Ismail, M. Inferring degrees from incomplete networks and nonlinear dynamics. *Int. Jt. Conf. Artif. Intell.* **29**, 3307–3313 (2020).
16. Zhang, Y., Shao, C., He, S. & Gao, J. Resilience centrality in complex networks. *Phys. Rev. E* **101**, 022304 (2020).
17. Zhang, H., Wang, Q., Zhang, W., Havlin, S. & Gao, J. Estimating comparable distances to tipping points across mutualistic systems by scaled recovery rates. *Nat. Ecol. Evolution* **6**, 1524–1536 (2022).
18. Newman, M. E. Mixing patterns in networks. *Phys. Rev. E* **67**, 026126 (2003).
19. Holland, J. N., DeAngelis, D. L. & Bronstein, J. L. Population dynamics and mutualism: functional responses of benefits and costs. *Am. Naturalist* **159**, 231–244 (2002).
20. Wilson, H. R. & Cowan, J. D. Excitatory and inhibitory interactions in localized populations of model neurons. *Biophysical J.* **12**, 1–24 (1972).
21. Vaswani, A. et al. Attention is all you need. In *Advances in Neural Information Processing Systems* **30**, (NIPS, 2017).
22. Kipf, T. N. & Welling, M. Semi-supervised classification with graph convolutional networks. In *International Conferences on Learning Representations* **5**, (ICLR, 2017).
23. Li, H. et al. Inferring transcription factor regulatory networks from single-cell atac-seq data based on graph neural networks. *Nat. Mach. Intell.* **4**, 389–400 (2022).
24. Chen, S. & Jung, Y. A generalized-template-based graph neural network for accurate organic reactivity prediction. *Nat. Mach. Intell.* **4**, 772–780 (2022).
25. Wang, Y., Wang, J., Cao, Z. & Barati Farimani, A. Molecular contrastive learning of representations via graph neural networks. *Nat. Mach. Intell.* **4**, 279–287 (2022).
26. Zhang, J. et al. Predicting unseen antibodies' neutralizability via adaptive graph neural networks. *Nat. Mach. Intell.* **4**, 964–976 (2022).
27. Prasse, B. & Van Mieghem, P. Predicting network dynamics without requiring the knowledge of the interaction graph. *Proc. Natl Acad. Sci.* **119**, e2205517119 (2022).
28. Xu, K., Hu, W., Leskovec, J. & Jegelka, S. How powerful are graph neural networks? In *International Conferences on Learning Representations* **7**, (ICLR, 2019).
29. Hu, J., Amor, D. R., Barbier, M., Bunin, G. & Gore, J. Emergent phases of ecological diversity and dynamics mapped in microcosms. *Science* **378**, 85–89 (2022).
30. Takeuchi, Y. & Adachi, N. Stable equilibrium of systems of generalized volterra type. *J. Math. Anal. Appl.* **88**, 157–169 (1982).
31. Goldford, J. E. et al. Emergent simplicity in microbial community assembly. *Science* **361**, 469–474 (2018).
32. Hochreiter, S. & Schmidhuber, J. Long short-term memory. *Neural Comput.* **9**, 1735–1780 (1997).
33. Borgwardt, K. M. & Kriegel, H.-P. Shortest-path kernels on graphs. *IEEE Int. Conf. Data Min.* **5**, 8 (2005).
34. Neumann, M., Garnett, R., Bauckhage, C. & Kersting, K. Propagation kernels: efficient graph kernels from propagated information. *Mach. Learn.* **102**, 209–245 (2016).
35. Interaction web database. Data are available at http://www.ecologia.ib.usp.br/iwdb/html/resources.html.
36. Erdős, P. et al. On the evolution of random graphs. *Publ. Math. Inst. Hungarian Acad. Sci.* **5**, 17–60 (1960).
37. Barabási, A.-L. Network science. *Philos. Trans. R. Soc. A: Math., Phys. Eng. Sci.* **371**, 20120375 (2013).
38. Sanhedrai, H. et al. Reviving a failed network through microscopic interventions. *Nat. Phys.* **18**, 338–349 (2022).
39. Levin, S. A. Self-organization and the emergence of complexity in ecological systems. *Bioscience* **55**, 1075–1079 (2005).
40. Moffett, K. B., Nardin, W., Silvestri, S., Wang, C. & Temmerman, S. Multiple stable states and catastrophic shifts in coastal wetlands: Progress, challenges, and opportunities in validating theory using remote sensing and other methods. *Remote Sens.* **7**, 10184–10226 (2015).
41. Pastor-Satorras, R., Castellano, C., Van Mieghem, P. & Vespignani, A. Epidemic processes in complex networks. *Rev. Mod. Phys.* **87**, 925 (2015).
42. Wodarz, D., Christensen, J. P. & Thomsen, A. R. The importance of lytic and nonlytic immune responses in viral infections. *Trends Immunol.* **23**, 194–200 (2002).
43. Shandilya, S. G. & Timme, M. Inferring network topology from complex dynamics. *N. J. Phys.* **13**, 013004 (2011).
44. Newman, M. E. Network structure from rich but noisy data. *Nat. Phys.* **14**, 542–545 (2018).
45. Zhou, H. et al. Informer: Beyond efficient transformer for long sequence time-series forecasting. *AAAI Conf. Artif. Intell.* **35**, 11106–11115 (2021).
46. Zhou, T. et al. Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. *Int. Conf. Mach. Learn.* **39**, 27268–27286 (2022).
47. Wu, H., Xu, J., Wang, J. & Long, M. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Adv. Neural Inf. Process. Syst.* **34**, 22419–22430 (2021).
48. Li, Q., Han, Z. & Wu, X.-M. Deeper insights into graph convolutional networks for semi-supervised learning. In *The AAAI Conference on Artificial Intelligence* **32**, (AAAI, 2018).
49. Murphy, C., Laurence, E. & Allard, A. Deep learning of contagion dynamics on complex networks. *Nat. Commun.* **12**, 1–11 (2021).
50. Zimmerman, R. D., Murillo-Sánchez, C. E. & Thomas, R. J. Matpower: Steady-state operations, planning, and analysis tools for power systems research and education. *IEEE Trans. Power Syst.* **26**, 12–19 (2010).
51. Meyur, R. et al. Ensembles of realistic power distribution networks. *Proc. Natl Acad. Sci.* **119**, e2205772119 (2022).
52. García-Girón, J. et al. Shifts in food webs and niche stability shaped survivorship and extinction at the end-cretaceous. *Sci. Adv.* **8**, eadd5040 (2022).
53. Sanchez-Gonzalez, A. et al. Learning to simulate complex physics with graph networks. *Int. Conf. Mach. Learn.* **37**, 8459–8468 (2020).
54. Das, P. et al. Accelerated antimicrobial discovery via deep generative models and molecular dynamics simulations. *Nat. Biomed. Eng.* **5**, 613–623 (2021).
55. Zhong, L., Lopez, D., Pei, S. & Gao, J. Healthcare system resilience and adaptability to pandemic disruptions in the United States. *Nat Med.* **30**, 2311–2319 (2024).
56. Devlin, J., Chang, M., Lee, K. & Toutanova, K. BERT: pre-training of deep bidirectional transformers for language understanding. *2019 Conf. North Am. Chapter Assoc. Computational Linguist.: Hum. Lang. Technol.* **1**, 4171–4186 (2019).
57. Ba, J. L., Kiros, J. R. & Hinton, G. E. Layer normalization. Preprint at https://arxiv.org/abs/1607.06450 (2016).
58. Woo, S., Park, J., Lee, J.-Y. & Kweon, I. S. Cbam: Convolutional block attention module. *Eur. Conf. Computer Vis.* **15**, 3–19 (2018).
59. Kingma, D. P. & Ba, J. Adam: a method for stochastic optimization. In *International Conferences on Learning Representations* **3**, (ICLR, 2015).
60. Morone, F., Del Ferraro, G. & Makse, H. A. The k-core as a predictor of structural collapse in mutualistic ecosystems. *Nat. Phys.* **15**, 95–102 (2019).
61. Dormand, J. R. & Prince, P. J. A family of embedded runge-kutta formulae. *J. Computational Appl. Math.* **6**, 19–26 (1980).
62. Balaji, S., Babu, M. M., Iyer, L. M., Luscombe, N. M. & Aravind, L. Comprehensive analysis of combinatorial regulation using the transcriptional regulatory network of yeast. *J. Mol. Biol.* **360**, 213–227 (2006).

63. Gama-Castro, S. et al. Regulondb (version 6.0): gene regulation model of escherichia coli k-12 beyond transcription, active (experimental) annotated promoters and textpresso navigation. *Nucleic Acids Res.* **36**, D120–D124 (2008).
64. Wilson, H. R. & Cowan, J. D. A mathematical theory of the functional dynamics of cortical and thalamic nervous tissue. *Kybernetik* **13**, 55–80 (1973).
65. Karlebach, G. & Shamir, R. Modelling and analysis of gene regulatory networks. *Nat. Rev. Mol. Cell Biol.* **9**, 770–780 (2008).
66. Liu, C. et al. Datasets and codes of deep learning resilience inference for complex networked systems. https://doi.org/10.5281/zenodo.13742983 (2024).

Acknowledgements

This work was supported in part by the National Key Research and Development Program of China under grant 2022ZD0116402, National Natural Science Foundation of China under grant 23IAA02114 and U22B2057, and Tsinghua University-Toyota Research Center to Y.L. and F.X., and the USA National Science Foundation under No. 2047488 to J.G. The funders had no role in study design, data collection and analysis, decision to publish or preparation of the manuscript.

Author contributions

F.X., Y.L., and J.G. jointly launched this research and provided the research outline. C.L., F.X., C.G., and Y.L. designed the research methods. C.L. performed the experiments and prepared the figures. F.X., C.G., Z.W., Y.L. and J.G. provided critical revisions. All authors jointly analyzed the results and participated in the writing of the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41467-024-53303-4.

Correspondence and requests for materials should be addressed to Yong Li or Jianxi Gao.

Peer review information *Nature Communications* thanks the anonymous reviewers for their contribution to the peer review of this work. A peer review file is available.

Reprints and permissions information is available at http://www.nature.com/reprints

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by-nc-nd/4.0/.

© The Author(s) 2024